

Zdalny system pomiarowy z interfejsem Ethernet, część 2

AVT-5111

Układy monitoringu i zdalnego sterowania zyskały nową jakość wraz z upowszechnieniem szerokopasmowego dostępu do Internetu. Podobnie stało się z rozproszonymi systemami pomiarowymi pozwalającymi sterować nawet złożonymi procesami produkcyjnymi z poziomu jednej aplikacji. W artykule prezentujemy przykładowy projekt systemu pomiarowego, w którym sterowanie odbywa się poprzez Internet.

Rekomendacje: projekt mimo sporych możliwości i dość rygorystycznych założeń technicznych stanowi doskonale pole doświadczalne dla projektantów właśnie ujarzmiających Ethernet.



Komunikacja między modułami. Interfejs RS485

W środowisku przemysłowym bardzo dużą popularnością cieszy się magistrala RS485. Jest ona łatwa do zaimplementowania, zapewnia dużą odporność na zakłócenia oraz dużą maksymalną długość, nie wymaga także kosztownego okablowania.

Budowa standardowych układów nadajnika/odbiornika RS485 pozwala na podłączenie do wspólnej magistrali maksymalnie 32 urządzeń. Parametr ten limituje więc liczbę modułów składających się na system pomiarowy. Zresztą trudno tu mówić o jakimś limicie, wystarczy sobie uzmysłowić, że 32-modułowy system miałby wtedy 128 wejść analogowych, a taka liczba przekracza typowe wymagania dotyczące systemów pomiarowych. Przykładowe połączenie kilku modułów wygląda jak na rys. 6.

Standard EIA-485 jest dzisiaj najbardziej rozpowszechnionym wielopunktowym i różnicowym interfejsem elektrycznym. Tym, co wyróżnia go od innych jest fakt, iż umożliwia dwukierunkową komunikację między urządzeniami przy użyciu jedynie pary przewodów. Powszechnie używa się do tego celu skrętki (np. takiej, jaką wykorzystują sieci komputerowe). Maksymalny zasięg magistrali określa się na 1200 m, natomiast prędkości uzyskiwane (ale na mniejszych odległościach) sięgają 10 Mbit/s. Elektryczne parametry typowych transceiverów ograniczają liczbę możliwych do podłączenia

urządzeń (końcówek) do 32. Wynika to z limitu obciążalności prądowej nadajników.

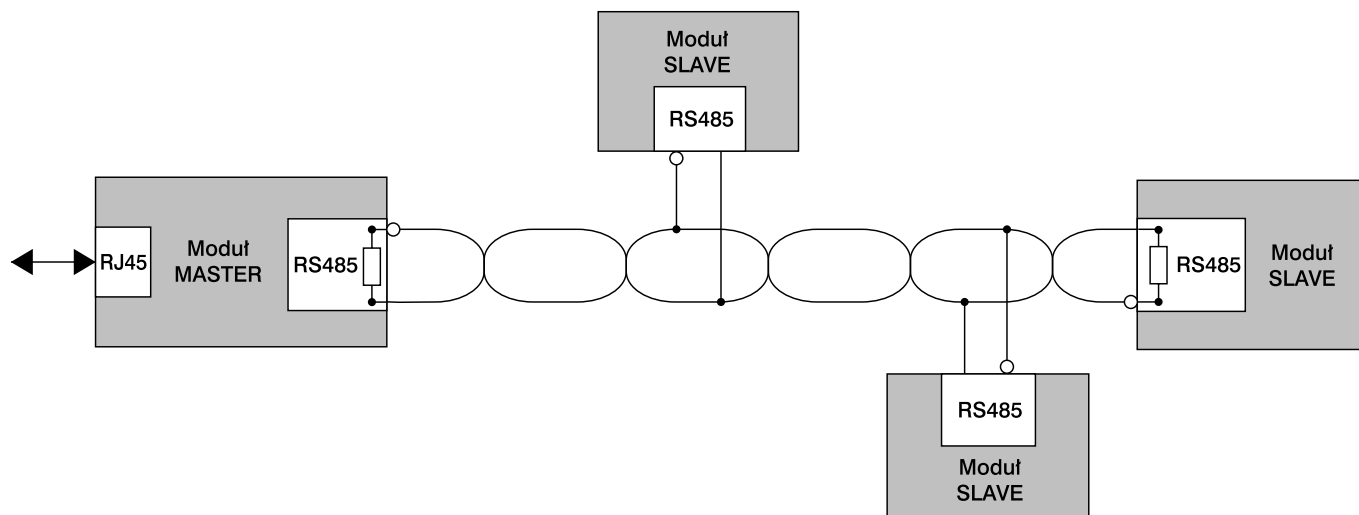
Różnicowy sposób przesyłania danych polega na sterowaniu linii B przebiegiem będącym w przeciwfazie do sygnału na linii A. Zakłócenia, które na dużych odległościach indukują się w obu przewodach mają mniej więcej ten sam poziom, a odbiornik, wykonując odejmowanie przebiegów linii A i B, powoduje że zakłócenia te znoszą się i są usuwane z przebiegu wyjściowego.

Standard EIA-485 definiuje jedynie normy elektryczne interfejsu. Nie mówi natomiast nic na temat protokołów transmisji, czy też standardu łącz. Tutaj już projektant ma wolną rękę i może dobrać odpowiednie rozwiązanie pasujące do jego sytuacji.

RS485 jest magistralą dwukierunkową mogącą pracować w trybie half-duplex. Oznacza to, że nie jest możliwe jednoczesne nadawanie i odbiór (do tego celu potrzebne by były dwie osobne pary). Jak już wcześniej wspomniano limit 32 urządzeń wynika z typowych parametrów transceiverów i wymagań standardu. Praca z większą liczbą końcówek grozi uszkodzeniem odbiornika i/lub nadajnika oraz pogorszeniem odporności magistrali na zakłócenia. Standard mówi o tym, że nadajniki muszą mieć wydajność prądową co najmniej 60 mA. Jako minimalną wartość napięcia różnicowego przyjmuje się 1,5 V. Magistrala powinna być też zakończona po obu stronach terminatorami

PODSTAWOWE PARAMETRY

- Płytki o wymiarach: 136x128 (151 z wypustkami) mm
- Podstawowe cechy systemu pomiarowego:
 - liczba modułów połączonych wspólną magistralą RS485: od 1 do 32,
 - liczba wejść/wyjść jednego modułu: 4 wejścia z pętlą prądową 0/4...20 mA, 2 wejścia stykowe oraz 2 wyjścia przekątnikowe,
 - możliwość zapisywania kolejnych pomiarów w wewnętrznej pamięci EEPROM (w każdym module),
 - stempel czasowy dodawany do każdego pomiaru,
 - możliwość odczytu mierzonych wartości na wbudowanym wyświetlaczu,
 - zabezpieczenie zapisanych parametrów i nastaw zegara czasu rzeczywistego przed zanikami napięcia zasilania,
 - zdalny podgląd stanu czujników oraz ustawianie wyjść przekątnikowych,
 - obsługa systemu przez aplikację uruchomioną na komputerze PC



Rys. 6. Przykładowe połączenie kilku modułów na magistrali RS485

(czyli obciążona rezystorami o wartości równej impedancji falowej linii transmisyjnej, dla skrętki komputerowej jest to ok. 100 Ω). Prąd płynący przez te rezystory będzie więc miał natężenie 30 mA. Prąd wejściowy odbiornika jest definiowany na ok. 1 mA. Zsumowanie prądu wejściowego najgorszego przypadku dla 31 odbiorników z prądem płynącym przez terminatory linii daje wspomniane wcześniej 60 mA. Przekroczenie tej granicy jest już traktowane jako naruszenie standardu. Istnieją wprawdzie odbiorniki magistrali RS485 o obniżonym prądzie wejściowym, ale dzieje się tak kosztem maksymalnej szybkości transmisji.

Zalecanym sposobem podłączenia urządzeń do magistrali jest sposób łańcuchowy (*Daisy-Chain*). Należy unikać konfiguracji typu „gwiazda”. Linia transmisyjna powinna mieć wyraźnie zaznaczony początek i koniec. W miejscach tych powinny być umieszczone terminatory. Wykonany prototyp umożliwia włączenie bądź odłączenie terminatora, który jest obecny w każdym module, za pomocą zwykłej zworki.

Istnieje kilka metod podłączania terminatorów do linii transmisyjnej. Najprostsza to użycie na obu końcach rezystora o wartości ok. 100 Ω (dla skrętki) równoległe do linii. Jednak metoda ta jest nie do zaakceptowania w przypadku RS485. Jak już to zostało wcześniej wspomniane, magistrala ta jest wielopunktowa. Każde urządzenie do niej podłączone może zarówno odbierać, jak i nadawać dane. Zastanówmy się teraz, jaki stan panuje na linii,

gdy żaden z nadajników nie nadaje. Napięcie różnicowe linii jest w takiej sytuacji bliskie zeru, a w związku z tym, stan na wyjściu odbiorników jest nieokreślony. Jest to bardzo niebezpieczna sytuacja, mogąca skutecznie uniemożliwić jakąkolwiek komunikację. Przy transmisji asynchronicznej start transmisji jest zwykle wykrywany przez pojawienie się zbocza, sygnalizującego początek ramki. Stan nieokreślony na wyjściu odbiornika może powodować fałszywe wyzwolenia i sygnalizację błędów ramki. Aby takich sytuacji uniknąć konieczne jest ustalenie jakiegoś zdefiniowanego stanu linii w momencie, gdy żaden z nadajników nie jest aktywny. Wstępną polaryzację, powodującą powstanie na linii w stanie spoczynku napięcia różnicowego, uzyskuje się przez dołączenie z jednej ze stron magistrali dwóch dodatkowych rezystorów, co pokazano na rys. 7.

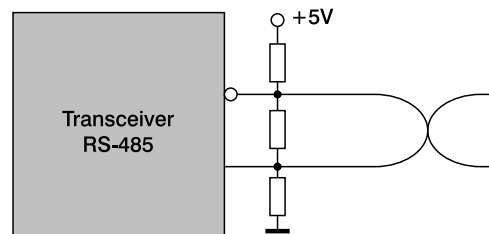
Pomimo faktu, iż praca różnicowa nie wymaga dodatkowego przewodu masy, w praktyce okazuje się, że zastosowanie dodatkowego przewodu, będącego wspólnym odniesieniem do potencjału masy, jest niezbędne. Wynika to z dopuszczalnego zakresu napięć wspólnych transceiverów. Napięcie wspólne określa się jako wartość średnią napięć na dwóch wyjściach nadajnika, bądź też dwóch wejść odbiornika. Standard określa ten parametr na zakres od -7 V do +12 V. Taka różnica potencjałów między nadajnikami/odbiornikami jeszcze nie powoduje ich uszkodzenia. Duża różnica tych potencjałów może w szcze-

gólności wystąpić przy dużych odległościach między poszczególnymi końcówkami. Przyczyny ich powstawania to różnica między potencjałami uziemienia, nakładanie się zakłóceń, napięcie offsetu nadajnika i inne. Brak wspólnego przewodu masy może również w bardzo znaczący sposób podnieść poziom generowanych zakłóceń elektromagnetycznych (EMI). W przypadku interfejsu dwuprzewodowego prąd wspólny nie ma niskoimpedancyjnej drogi powrotu, co powoduje wzrost emisji zakłóceń do otoczenia.

Remedium na te niekorzystne efekty jest zastosowanie właśnie przewodu masy, razem z dwoma przewodami różnicowymi. Często też, tego dodatkowego przewodu nie podłącza się bezpośrednio do masy nadajnika-odbiornika, a przez szeregowy rezystor. Powoduje to ograniczenie płynących tym przewodem prądów wspólnych.

Protokół transmisji

Wyboru protokołu transmisji należy dokonać niezależnie od standardu elektrycznego, jakim jest RS485. Można do tego celu wykorzystać istniejące protokoły stosowane w automatyce przemysłowej, bądź też



Rys. 7. Terminator magistrali RS485 ze wstępną polaryzacją

Tryb ASCII	START	1	2	3	4	5	6	7	STOP/ PAR	STOP
------------	-------	---	---	---	---	---	---	---	--------------	------

TRYB RTU	START	1	2	3	4	5	6	7	8	STOP/ PAR	STOP
----------	-------	---	---	---	---	---	---	---	---	--------------	------

Rys. 8. Format ramki pojedynczego znaku protokołu Modbus

opracować własny, definiujący sposób przesyłania komunikatów między modułami. Najpopularniejszymi protokołami stosowanymi w przemyśle, a wykorzystujące standard elektryczny RS485 są MODBUS oraz PROFIBUS wraz z odmianami. W opisywanym urządzeniu zdecydowano się na użycie uproszczonej wersji protokołu MODBUS. Ma to też taką zaletę, iż po niewielkiej modyfikacji kodu mikrokontrolerów modułów, możliwe jest dołączenie do systemu fabrycznych urządzeń używających tego protokołu.

Protokół MODBUS. MODBUS to protokół transmisji szeregowej opracowany w 1978 roku przez firmę Modicon w celu wymiany danych między sterownikami własnej produkcji. Z czasem rozprzecznił się i stał się *de facto* standardem wymiany danych w układach sterowników przemysłowych. Urządzenia w standardzie Modbus komunikują się w relacji Master – Slave, czyli jedno z urządzeń (Master) pełni rolę nadrzędną – inicjuje wymianę danych wysyłając rozkazy (*Requests*) do urządzeń Slave. Układy podrzędne, wykonują rozkazy urządzenia Master, ewentualnie wysyłając odpowiedź (*Response*). Podstawowa wersja protokołu Modbus ma dwie odmiany: Modbus ASCII i Modbus RTU. Istnieją późniejsze, unowocześnione wersje, jak Modbus Plus oraz Modbus TCP/IP (przystosowany do pracy z sieciami Ethernet).

Znaki w protokole są wysyłane w ramach o formacie identycznym jak RS232. Ułatwia to podłączenie magistral stosujących ten protokół

do układów UART stosowanych powszechnie w mikrokontrolerach. Pojedyncza ramka może wyglądać tak jak na **rys. 8**. Każdy rozkaz kierowany do urządzenia Slave ma następujący format:

- adres urządzenia Slave,
- kod funkcji,
- opcjonalny blok danych,
- blok kontroli poprawności danych

Odpowiedź wygląda następująco:

- adres urządzenia Slave,
- kod funkcji lub w przypadku błędu – kod błędu,
- opcjonalne dane,
- blok kontroli poprawności danych.

Urządzenia Slave mogą mieć adresy od 1 do 247. Adres 0 jest zarezerwowany dla komunikatów rozgłoszeniowych (*broadcast*), kierowanych do wszystkich urządzeń Slave. W tym ostatnim przypadku z oczywistych powodów nie jest przesyłana odpowiedź.

Tryb ASCII. W trybie ASCII (znakowym), każdy bajt wymaga przesłania dwóch znaków ASCII definiujących jego wartość w postaci heksadecymalnej. Dozwolone znaki to cyfry 0...9 oraz litery A...F. Każda wiadomość w tym trybie rozpoczyna się od znaku dwukropka „:” o kodzie ASCII 3Ah, natomiast jest kończona znakiem powrotu karetki i nowej linii (CRLF, kody ASCII – 0Dh oraz 0Ah). Format komunikatu przedstawiono na **rys. 9**. Cechą szczególną tego trybu jest to, iż dopiero przerwa między znakami przekraczająca 1 sekundę jest traktowana jako wystąpienie błędu transmisji.

START	ADRES	FUNKCJA	DANE	KONTROLA LRC	KONIEC
1znak :	2 znaki	2 znaki	n znaków	2 znaki	2 znaki CRLF

Rys. 9. Format ramki komunikatu w trybie ASCII

START	ADRES	FUNKCJA	DANE	KONTROLA LRC	KONIEC
T1-T2-T3-T4	8 bitów	8 bitów	n x 8 bitów	16 bitów	T1-T2-T3-T4

Rys. 10. Format ramki komunikatu w trybie RTU

Do kontroli poprawności przesyłanych komunikatów w trybie ASCII służy dwuznakowy ciąg kontrolny LRC (*Longitudinal Redundancy Check*). Wartość bajtu, będącego wynikiem kontroli LRC jest uzyskiwany na drodze obliczeniowej, polegającej na dodaniu do siebie kolejnych bajtów (reprezentujących znaki) komunikatu (z pominięciem początkowego znaku dwukropka oraz kończących znaków CRLF) ignorując wszelkie przeniesienia arytmetyczne, a następnie wykonaniu operacji uzupełnienia do dwóch wyniku. Obliczona wartość jest następnie dodawana do komunikatu po bloku danych. Odbiorca wiadomości, samodzielnie obliczając bajt LRC i porównując go z odebranymi znakami LRC, może stwierdzić wystąpienie błędu podczas transmisji.

Tryb RTU (*Remote Terminal Unit*). W trybie RTU każdy bajt przesyłanego komunikatu zawiera dwa 4-bitowe znaki heksadecymalne, czyli po prostu bajt jest przesyłany zwykłym kodem binarnym. Pozwala to na dwukrotne zmniejszenie długości komunikatu i jego szybsze przesłanie. Nie stosuje się w tym trybie żadnego bajtu startowego (jest to niemożliwe ze względu na binarny sposób transmisji), natomiast standard wymaga, aby komunikaty były oddzielone od siebie ciszą transmisyjną o długości co najmniej 3,5 znaku (4-bitowego). W praktyce najczęściej przyjmuje się długość odstępu na 4 znaki (2 bajty). Jeśli natomiast między przesyłanymi bajtami znajdzie się przerwa o długości powyżej 1,5 znaku, odbiornik porzuca odebrane wcześniej bajty i ponownie oczekuje na wystąpienie startowego bajtu adresującego. Postać ramki w tym trybie pokazano na **rys. 10**. Inny jest tym razem sposób kontroli poprawności komunikatu. W trybie RTU stosuje się cykliczną kontrolę nadmiarową CRC (*Cyclic Redundancy Check*). Jej algorytm jest nieco bardziej skomplikowany, nie będzie więc tutaj omawiany.

Funkcje i kody błędów protokołu Modbus

Kody najczęściej używanych funkcji wraz z ich przeznaczeniem przedstawiono w **tab. 3**.

Tab. 3. Zestawienie wybranych funkcji protokołu Modbus

Kod funkcji (dziesiętnie)	Nazwa	Opis
01	Read Coil Status	Odczytuje stan wyjść dwustanowych (jednego lub więcej)
02	Read Input Status	Odczytuje stan wejść dwustanowych (jednego lub więcej)
03	Read Holding Registers	Odczytuje zawartość 16-bitowych rejestrów stanu
04	Read Input Registers	Odczytuje zawartość 16-bitowych rejestrów wejściowych
05	Force Single Coil	Włącza lub wyłącza wybrane wyjście dwustanowe
06	Preset Single Register	Ustawia wybrany 16-bitowy rejestr stanu
07	Diagnostic	Funkcja diagnostyczna, żądanie wykonania testu
15	Force Multiple Coils	Włącza lub wyłącza wiele wyjść dwustanowych
16	Preset Multiple Registers	Ustawia kilka 16-bitowych rejestrów stanu

W razie wystąpienia błędu (np. niewłaściwe zapytanie), urządzenie Slave w odpowiedzi modyfikuje kod funkcji w ten sposób, że ustawia najstarszy bit w zwracanym kodzie funkcji, a w następnym bajcie (dla trybu RTU) umieszcza kod błędu. Możliwe kody przedstawiono w tab. 4.

Implementacja komunikacji szeregowej

Do komunikacji między modułami (interfejsem RS485) wybrano protokół Modbus RTU. Głównym powodem wybrania tego rozwiązania jest zwiększona przepustowość tej odmiany w porównaniu z trybem ASCII. Oczywiście wymiana pojedynczych komunikatów nie jest tutaj problemem, chodziło raczej o przyspieszenie procesu odczytu całej zawartości pamięci modułów podrzędnych. Proponowany rozmiar pamięci pojedynczego modułu to 1 Mbit, tak więc odczyt takiej ilości danych interfejsem szeregowym może potrwać nawet do kilkunastu sekund (w zależności od przyjętej prędkości transmisji). Komunikacja między modułami odbywa się z prędkością 76,8 kbit/s. W trybie ASCII czas ten ulega mniej więcej podwojeniu.

Urządzeniom Slave nadano adresy od 01 do 31. Każdy z modułów (także Master) wyposażono w scalony transceiver interfejsu RS485 – układ MAX485 (lub odpowiednik innego producenta), który został podłączony do mikrokontrolera modułu.

Mikrokontroler

Jako główny mikrokontroler modułów wybrano układ firmy Atmel – ATmega32 należący do rodziny AVR i mający 32 kB pamięci Flash. Mikrokontroler posiada 32 linie wejścia/wyjścia – wystarczającą liczbę do podłączenia wszystkich

układów i elementów peryferyjnych. Zawiera on pewne elementy architektury RISC, większość rozkazów wykonuje w jednym taktie zegara, co gwarantuje dużą szybkość wykonywania programu. Programowanie mikrokontrolera może odbywać się w docelowym układzie za pomocą specjalnego złącza ISP (*In-System Programming*). Niewątpliwą zaletą całej rodziny AVR jest istnienie darmowego kompilatora języka C – WinAVR. Tego kompilatora użył także autor.

Pamięć i zegar czasu rzeczywistego

Głównym zadaniem prawie każdego systemu pomiarowego jest zbieranie danych pomiarowych w celu ich późniejszej analizy. Jeśli mamy do czynienia z systemem czasu rzeczywistego, dane mogą być interpretowane na bieżąco (np. przez obsługę) i konieczność ich zbierania i zapamiętywania traci nieco na znaczeniu. Jednak w przeważającej części przypadków, pomiary są zapisywane w pamięci nieulotnej. Umożliwia to generowanie statystyk, tendencji itp., zapobiega też przecenieniu pewnych krótkotrwałych zjawisk mogących wystąpić w mierzonym obiekcie (oczywiście słowo „krótkotrwały” może mieć bardzo

różne znaczenie w zależności od obiektu).

Zbieranie danych ma też dodatkową zaletę w systemie komunikującym się przez sieć Ethernet – nie ma konieczności, by komputer zdalny był non stop podłączony do systemu. W momencie podłączenia pobiera on dane zaległe i dalej już na bieżąco wykonuje każdy nowy pomiar.

Wszystkie wymienione argumenty zdecydowały o umieszczeniu w każdym z modułów pamięci nieulotnej EEPROM. Jej wielkość zależy od częstotliwości pomiarów, wielkości (w bajtach) bloku danych opisujących jeden pomiar oraz wymaganym minimalnym okresem czasu, w jakim dane mają być zbierane i zapisywane. Autor zdecydował się na użycie układu AT24C1024. Jest to pamięć o pojemności 1 Mbit (128 k słów 8-bitowych) z interfejsem I²C. Do obsługi wykorzystano magistralę I²C, gdyż wymaga tylko 2 linii (SCL i SDA). Sama pamięć ma jedynie 8 wyprowadzeń, a zastosowany procesor ma wbudowany sprzętowy interfejs I²C co znacznie upraszcza procedury programowe.

Przyjmując 128 kB za wielkość bazową pamięci, 8 bajtów przeznaczonych na jeden pomiar, 6 kanałów mierzonych (4 kanały z pętlą prądową i 2 kanały stykowe), a pomiar co 10 sekund dla każdego kanału, pamięci wystarczy na ponad 450 minut ciągłych pomiarów.

W module Master zastosowano układ PCF8583 – zegar czasu rzeczywistego (RTC – *Real Time Clock*). Jest on, podobnie jak pamięć EEPROM, wyposażony w interfejs I²C. Zegar jest zabezpieczony przed utratą danych poprzez kondensator podtrzymujący o dużej pojemności (tzw. Backup Capacitor) 0,1 F. Układ charakteryzuje się bardzo niskim poborem prądu przy

Tab. 4. Zestawienie możliwych kodów błędów protokołu MODBUS

Kod błędu (dziesiętnie)	Nazwa	Opis
01	Illegal Function	Wybrana funkcja jest nieobsługiwana przez Slave'a
02	Illegal Data Address	Nieistniejący adres (np. rejestru)
03	Illegal Data Value	Niedozwolona wartość danych
04	Slave Device Failure	Wystąpił błąd podczas wykonywania polecenia
05	Acknowledge	Slave przyjął polecenie i je wykonuje, ale może zająć to dłuższy czas (zabezpieczenie przed przekroczeniem limitu czasu na odpowiedź)
06	Slave Device Busy	Slave zajęty wykonywaniem innego polecenia
07	Negative Acknowledge	Nie można wykonać polecenia (dotyczy funkcji 13 i 14 dziesiętnie – tutaj nie opisanych)
08	Memory Parity Error	Wykryto błąd parzystości pamięci

braku aktywności magistrali I²C – ok. 10 mA przy napięciu zasilającym 5 V. Kondensator z powodzeniem umożliwia nawet jednodniową pracę zegara w warunkach awarii zasilania sieciowego.

Układ zegara RTC w celu ograniczenia kosztów zastosowano tylko w module Master. Moduły Slave są synchronizowane z zegarem Mastera co kilkadziesiąt sekund, za pośrednictwem komunikatów rozgłoszeniowych protokołu Modbus. Mikrokontroler każdego z modułów Slave ma także wyznaczone jedno przerwanie licznikowe, zapewniające poprawne odliczanie czasu między kolejnymi komunikatami rozgłoszeniowymi.

Zasilanie

Każdy z modułów posiada własny zasilacz sieciowy z transformatorem o mocy 6 W. Moc pobierana przez elementy modułów nie jest duża, więc wielkość ta jest w zupełności wystarczająca. Transformator posiada dwa niezależne uzwojenia o nominalnym napięciu wyjściowym 6 VAC. Napięcie +5 V zasilające m.in. mikrokontroler jest wytwarzane w typowym układzie prostownika dwupółprzewodnikowego i stabilizatora liniowego. Drugie uzwojenie jest podłączone do układu powielacza napięcia o napięciu wyjściowym ok. +18 V. Napięcie ok. +23 V będące sumą obu napięć składowych jest podawane na stabilizator LM317, który z kolei wytwarza potrzebne napięcie +20 V.

Dosyć istotną rzeczą w przypadku zastosowanego powielacza napięcia jest dobór kondensatorów elektrolietycznych, bo to od ich pojemności w głównej mierze zależy zdolność zasilacza do utrzymania stabilnego napięcia +20 V przy zwiększonym poborze prądu. Napięcie na wejściu stabilizatora U11 nie powinno w takich warunkach spaść poniżej ok. 22 V (spadek napięcia na stabilizatorze LM317 przy prądzie 100 mA wynosi typowo ok. 1,8...2 V). Pobór prądu z gałęzi +20 V może wynosić w granicznym przypadku ok. 100 mA (4 pętle prądowe pobierające maksymalny prąd 20 mA oraz prąd przekaźnika).

Zabezpieczenie wejść i dobór elementów kanałów wejściowych

Głównymi elementami zabezpieczającymi zarówno wejścia pomiarowe, jak i wejścia/wyjścia różnicowe

transceivera RS485 są diody transil o napięciu nominalnym blokowania ok. 6 V, P6KE6.8CA. Transile o tak niskich napięciach charakteryzują się dość dużym prądem upływu w okolicach napięcia 5 V, więc mogłyby one powodować niewielki błąd linii pomiarowych A1...A4. Jednak ponieważ w warunkach normalnej pracy, przy prądzie pętli równym 20 mA, napięcie na diodzie transil nie przekracza 2 V, prąd upływu nie ma praktycznie żadnego znaczenia – dla górnej wartości prądu pętli 20 mA, czyli napięcia równego 2 V, błąd wnoszony przez transil wynosi w przybliżeniu 0,02%.

Przed niekorzystnymi efektami zwarcia przewodów pętli prądowej chroni dodatkowo bezpiecznik półprzewodnikowy R13 o prądzie zadziałania 100 mA. Jest on wspólny dla kanałów A1 do A4.

Rezystory szeregowo R15, R19, R25, R28 są dodatkowym elementem ograniczającym prąd w przypadku zwarcia przewodów pętli prądowej. Dość istotne jest, aby miały one małą wartość (w tym przypadku jest to 100 Ω), aby dodatkowo nie zmniejszać dostępnej różnicy potencjałów pętli. W przypadku przeciążeń napięciowych (impulsy wysokiego napięcia) na rezystorze tym odkłada się prawie cała część napięcia udarowego. Z tego też powodu istotna jest sama wartość rezystancji jak i dopuszczalna moc rozpraszana. Można wyliczyć, że przy pojedynczych impulsach napięcia 400 V (powtarzanych min. co 1 s), średnia moc rozpraszana nie przekroczy wartości 0,6 W, gdy impuls będzie krótszy od ok. 0,6 ms. W przypadku dłuższych impulsów udarowych może dojść do spalenia rezystora zabezpieczającego i należy się z tym liczyć. Zwarcie przewodów linii spowoduje zadziałanie bezpiecznika polimerowego i ograniczenie prądu w obwodzie do niskiej wartości. Usunięcie zwarcia powoduje powrót do normalnej pracy układu.

Rezystory 100 Ω, o dokładności 0,1%, będące właściwym przetwornikiem prąd/napięcie, mają maksymalną moc traconą 0,6 W, jednak ze względu na podwyższoną dokładność zalecane jest nie przekraczanie wartości 0,25 W (podwyższona temperatura może pogorszyć dokładność). W warunkach normalnej pracy, czyli przy prądzie do 20 mA, spadku napięcia do 2 V, moc tracona w tych rezystorach nie przekroczy 40 mW.

Układ zerowania

Rolę nadzorowania napięcia zasilania pełni układ MAX1232 generujący ponadto sygnał zerowania dla mikrokontrolera ATmega32 oraz modułu ethernetowego EM100 (ETH1). Dodatkowo, wyprowadzenie 7 (STB) tego układu jest dołączone do wyprowadzenia 4 (L1) modułu ETH1. Układ MAX1232 ma bowiem także wbudowany układ watchdoga – co pewien czas na wyprowadzeniu STB musi zostać podany impuls kasujący wewnętrzny licznik watchdoga. Gdy czas ten zostanie przekroczony zostaje wygenerowany sygnał zerowania. Impulsy te generowane są właśnie przez moduł ETH1. Zabezpiecza to przed zawieszeniem się modułu EM100.

Wyjścia przekaźnikowe

Jak już to wcześniej zostało wspomniane, wyjście C1 jest sterowane przekaźnikiem elektromechanicznym, natomiast wyjście C2 przekaźnikiem półprzewodnikowym. Styki przekaźnika elektromechanicznego mają obciążalność 6 A/250 VAC.

Przekaźnikowi U4 należy się nieco więcej uwagi. Zawiera on w swojej strukturze optyczny układ separujący oraz triak będący elementem wykonawczym. Nominalny prąd układu to 2 A. Ze względu na tyrystorowy charakter wyjścia, do podtrzymania przepływu prądu obciążenia niezbędny jest pewien prąd progowy (prąd podtrzymania), który dla zastosowanego elementu S202T02 jest określony na 25 mA (maksymalnie). Oznacza to, że aby poprawnie sterować urządzeniem podłączonym do wyjścia C2, musi być to element o poborze mocy minimum 5,75 W (dla napięcia sieciowego 230 V).

Należy również pamiętać, że element półprzewodnikowy jest dużo bardziej wrażliwy na udary (prądowe lub napięciowe), które mogą spowodować jego uszkodzenie. Jednym z zabezpieczeń jest warystor V1 o napięciu ograniczania 400 V. Przed udarem prądowym niestety nie ma prostej ochrony.

Na wyjściu przekaźnika U4 zastosowano także układ gaszący C12, R11. Jego głównym zadaniem jest niedopuszczenie do niezamierzonego wyzwolenia wewnętrznego triaka, spowodowanego zbyt dużą szybkością narastania prądu (napięcia) obciążenia. Sytuacja taka może wystąpić w przypadku obciążeń o indukcyjnym lub pojemnościowym charakterze. Układ

Tab. 5. Wymagane ustawienia modułu EM100

Parametr	Ustawienie
Transport protocol	1 – TCP
RTS/CTS flow control	0 – Disabled or remote

gaszący ma za zadanie ograniczenie do bezpiecznej wartości wspomnianej szybkości narastania. Niestety nie ma prostej recepty na idealny układ gaszący; jest to kompromis między skutecznością działania z obciążeniem indukcyjnym lub pojemnościowym. Zwykle parametry RC układu dobiera się pod konkretne obciążenie. Wybrane przez autora wartości są tylko pewnym mniej lub bardziej udanym kompromisem.

Wyświetlacz alfanumeryczny

Zastosowany wyświetlacz został wykonany w technologii PLED, charakteryzującej się bardzo dobrą widocznością, szerokim kątem widzenia (ok. 160°) i małym poborem mocy. Typowo wyświetlacze te cechują się poborem prądu rzędu 20 μ A/piksel, co w typowych warunkach sprowadza

się do całkowitego poboru prądu ok. 10...15 mA. Oczywiście w przypadku tych wyświetlaczy nie trzeba stosować żadnego podświetlenia, które w przypadku tradycyjnych wyświetlaczy LCD pobierały stosunkowo duże prądy (ok. 100...200 mA).

Pozostałe uwagi

Mikrokontroler ATmega32 fabrycznie, za pomocą wewnętrznych bezpieczników programowalnych, jest dostarczany z włączonym oscylatorem wewnętrznym o częstotliwości ok. 1 MHz oraz włączonym interfejsem JTAG. W zaprojektowanym urządzeniu mikrokontroler pracuje z zewnętrznym kwarem 11,0592 MHz, więc konieczne jest przełączenie bezpieczników CKSEL3...0. W modułach linii wykorzystywane przez JTAG były używane do innych celów, przełączono także bezpiecznik JTAGEN. Włączono też układ nadzoru napięcia zasilania (*Brown-Out Detector*) – do tego celu służą bezpieczniki BODLEVEL i BODEN.

Do podłączenia programatora mikrokontrolera służy złącze JP2. Umoż-

liwia ono programowanie w docelowym układzie. Autor używał programu PonyProg2000 i prostego kabla z łączem równoległym (STK 200) podłączanym do komputera PC.

Zworka JP3 służy do włączenia rezystora terminującego magistralę RS485. W zależności od konfiguracji sieci interfejsowej należy odpowiednio ustawić tę zworkę tak, aby rezystory były włączone jedynie dla skrajnych dwóch modułów na magistrali.

Po pierwszym uruchomieniu modułu należy wyregulować odpowiednio kontrast wyświetlacza potencjometrem R30. Do poprawnej pracy systemu konieczne jest jeszcze skonfigurowanie modułu EM100. W tym celu należy posłużyć się aplikacją DS Manager, która można pobrać ze strony <http://www.tibbo.com>. Moduł Master należy podłączyć do tego samego segmentu sieci co komputer z uruchomioną aplikacją. Następnie konieczne jest ustawienie podanych w **tab. 5** parametrów (różniących się od domyślnych).

Andrzej Piernikarczyk
andrzej.piernikarczyk@gmail.com

LEMI-BIS

ul. Grabiszyńska 240
53-235 Wrocław

tel. (0-71) 339 00 29
339 00 30

faks (0-71) 339 05 01

lemibis@lemi.pl

złącza HDC

złączki listwowe

przyciski sterownicze

przełączniki elektromagnetyczne

SSR

przełączniki czasowe

czujniki indukcyjne i pojemnościowe

czujniki fotoelektryczne

regulatory temperatury PID

impulsowe zasilacze przemysłowe

www.lemi.pl

SKLEP INTERNETOWY 24h

POSZUKUJEMY DYSTRYBUTORÓW LOKALNYCH
 DOSKONAŁE WARUNKI HANDLOWE
 DUŻE RABATY

SPRZEDAŻ PEŁNEGO ASORTYMENTU Z MAGAZYNU • NAJLEPSZE CENY NA RYNKU

PDW MARTEL

WIĘCEJ NIŻ PROFESJONALNA
DYSTRYBUCJA

www.marthel.pl

PDW MARTEL
 ul. Sosnowa 24-5
 Bielany Wrocławskie
 55-040 Kobierzyce
 tel. +48 71 3110711, 12
 fax +48 71 3110713

Układy dźwiękowe serii ISD15000 firmy Winbond

W ofercie nowa seria układów dźwiękowych nagrywająco-odtwarzających **ISD15000**, zapewniających znakomitą jakość nagrywania i odtwarzania dźwięku. Zastosowano w nich przetwarzanie analogowych sygnałów audio do postaci cyfrowej, kompresję cyfrową, zapis do pamięci flash, kompleksowe zarządzanie pamięcią oraz zintegrowane ścieżki analogowych/cyfrowych sygnałów audio.

- częstotliwość próbkowania do 48 kHz
- stosunek sygnał/szum odpowiada rozdzielczości 12-bitowej
- algorytmy kompresji: ADPCM, PCM bez kompresji, kompresja typu μ
- interfejs I²S do szybkiego zapisu cyfrowego sygnału audio
- interfejs SPI dla cyfrowego sygnału audio i sterowania
- wejście analogowe dla sygnału audio
- różnicowe wejście mikrofonowe
- wyjściowy driver sygnału audio
- bezpośrednie wyjścia głośnikowe PWM
- regulacja głośności odtwarzanego dźwięku
- czas trwania nagrań przy 4-bitowej modulacji ADPCM i częstotliwości próbkowania 8 kHz:

ISD15001 - 1 min
ISD15002 - 2 min
ISD15004 - 4 min
ISD15008 - 8 min
ISD15016 - 16 min
ISD15032 - 32 min

- napięcie zasilania 2,7...3,6 V
- zakres temperatury pracy: komercyjny i przemysłowy