

Czytnik bezstykowych kart Unique



Projektowałem system kontroli dostępu, mający pracować w pomieszczeniach o dużym zapyleniu. Wymusiło to zastosowanie jako identyfikatorów elementów bezstykowych. Obecnie najłatwiej są dostępne karty zbliżeniowe o wymiarach standardowej karty plastikowej ISO. Pewną barierą była jednak dość wysoka cena czytników takich kart.

Rekomendacje: polecamy wszystkim Czytelnikom zainteresowanym tematyką bezstykowej kontroli dostępu.

Z pomocą przyszyły mi zamieszczone w EP reklamy zestawu ewaluacyjnego RFID, składającego się ze specjalizowanego układu ASIC oraz zestawu kart rozprawdzanych przez firmę Elatec z Krakowa. Okazało się to strzałem w dziesiątkę, gdyż nawet w przypadku (zazwyczaj kosztownego) prototypu koszt elementów nie przekroczył 100 zł.

W zbudowanym przeze mnie systemie znajduje się kilka czytników kart, które są sterowane przez centralkę (niebędącą przedmiotem niniejszego artykułu), która podejmuje decyzję o dostępie lub jego odmowie.

Sam czytnik, w wersji oprogramowania prezentowanego w artykule, nie zawiera żadnych mechanizmów decyzyjnych - jego zadania sprowadzają się do odczytania kodu zawartego na karcie, przekazania go na żądanie centralki oraz sterowania zamkiem elektromagnetycznym, również zgodnie z poleceniami centralki.

Magistralę systemową wykonano w oparciu o specyfikację elektryczną interfejsu RS485, a sposób przesyłania danych jest identyczny jak w przypadku RS232. Jednostka centralna jest jednocześnie kontrolerem magistrali i jako jedyna ma prawo zainicjować transmisję (wyjątkiem są uszkodzenia zgłaszane przez poszczególne urządzenia).

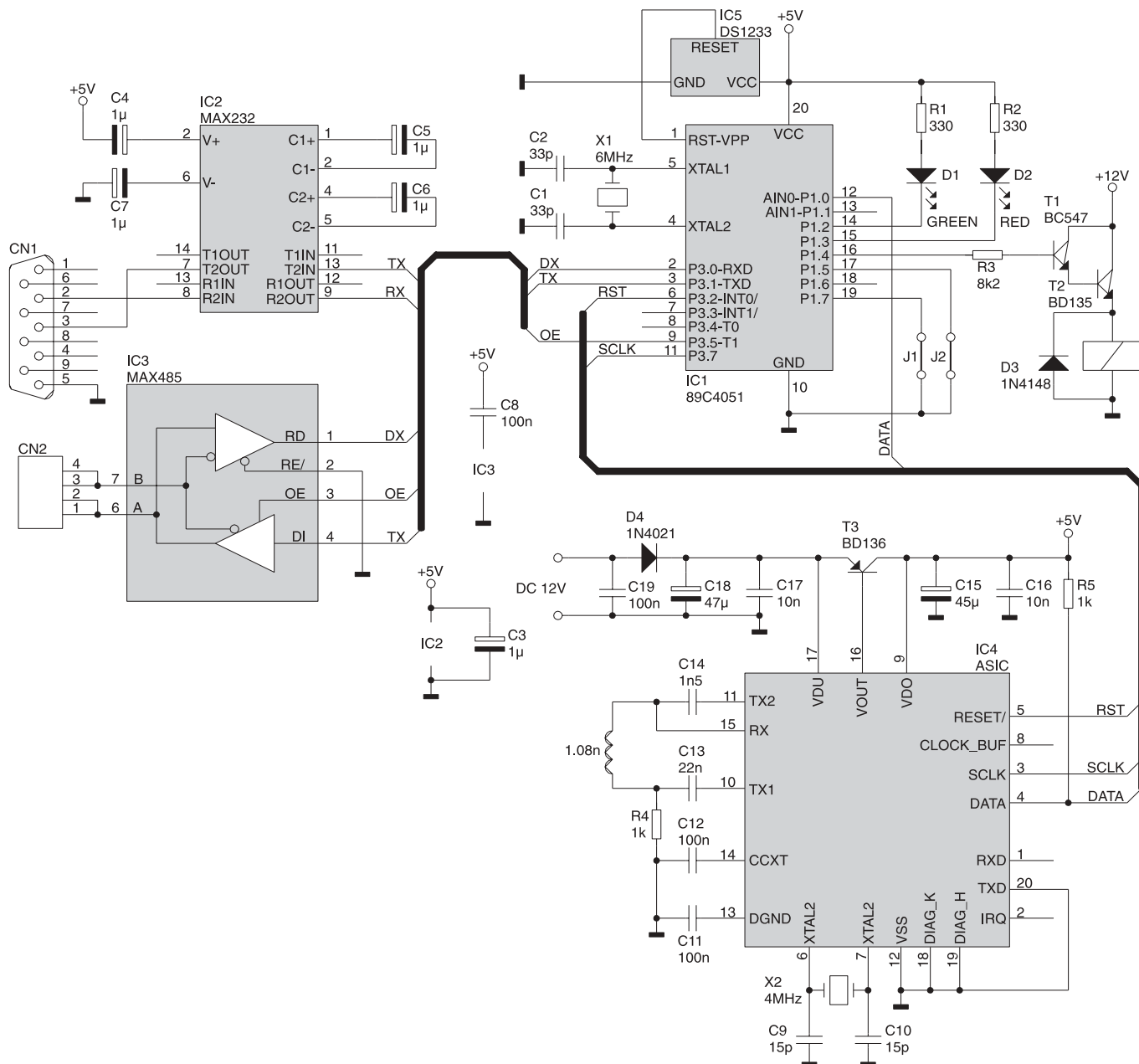
Każdy element systemu ma swój unikalny adres z zakresu 1...9 (0 jest adresem kontrolera). W czytniku można go ustawiać za pomocą zworek J1 i J2 (ustawiona binarnie wartość będzie dodawana do stałej ADR_OFFSET). W celu zwiększenia uniwersalności (jak również w celach testowych) układ czytnika wyposażono również w interfejs RS232C.

Wymienione założenia zdeterminowały sposób współpracy z czytnikiem, który opisano w dalszej części artykułu. Oczywiście, zmiana sposobu działania (np. wysyłanie kodu karty natychmiast po jego odczytaniu) jest jak najbardziej możliwa, co więcej, nieskomplikowana ze względu na modułową budowę programu czytnika.

Opis układu

Układ składa się z kilku bardzo wyraźnie rozdzielonych (co widać na schemacie - **rys. 1**) modułów:

- Czytnik, wykonany na układzie ASIC, będący poprawioną wersją układu z noty aplikacyjnej.
- Układu mikrokontrolera i jego najbliższych „peryferii“ (diod świecących i sterowania przełącznikiem), całkowicie standardowego i niewymagającego szczegółowego opisu. Jedyne zastosowanie specjalistycznego układu generowania zerowania może budzić zdziwienie, ale jest wynikiem niechęci autora projektu do kombinacji kondensator-opornik, wykazującej „złośliwość“ i niechęć do prawidłowego działania w warunkach niestabilnego zasilania.



Rys. 1. Schemat elektryczny

- Dwóch równoważnych układów interfejsu szeregowego, pozwalających wybrać (przez osadzenie odpowiedniego układu drivera) transmisję via RS232C lub RS485.

Karty Unique

Bezstykowe karty Unique są najprostszym funkcjonalnie przedstawicielem rodziny kart zbliżeniowych. Zawierają w sobie wyłącznie 40-bitowy, unikalny numer. Zawarty jest on w programowanej na etapie produkcji karty 64-bitowej pamięci ROM o następującej organizacji (rys. 2):

- 9 bitów nagłówka (same jedynki),

- 40 bitów numeru karty Dyx (z czego pierwszych 12 stanowi identyfikator nadawany konkretnemu klientowi przez producenta), podzielonego na 10 wierszy po 4 bity,
- 10 bitów parzystości wierszy PRy (występują po każdym z wierszy numeru),
- 4 bity parzystości kolumn PCx (po ostatnim wierszu i jego bicie parzystości),
- 1 bitu stopu (zero).

Po wprowadzeniu karty w pole magnetyczne czytnika (które w zależności od zastosowanej w nim anteny może mieć zasięg do 50 cm), w cewce wbudowanej w kartę zostaje zaindukowane napięcie

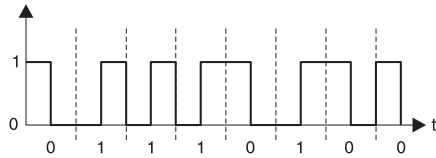
zasilające. Z chwilą, gdy osiągnie ono wymagany poziom, karta rozpoczyna działanie, polegające na cyklicznym wysyłaniu całej zawartości pamięci (po wysłaniu bitu stopu transmisja zaczyna się ponownie, bez żadnej przerwy). Dane są kodowane kodem Manchester, który moduluje metodą ASK amplitudę nośnej o częstotliwości 125 kHz. Wysłanie jednego bitu danych zajmuje 64 okresy fali nośnej, co daje czas 512 μ s.

Kod Manchester

Kodowanie Manchester (nazwa pochodzi od uniwersytetu, gdzie zostało opracowane) pozwala na zintegrowanie strumienia danych

1	1	1	1	1	1	1	1	1	1
D00	D01	D02	D03	PR0					
D10	D11	D12	D13	PR1					
D20	D21	D22	D23	PR2					
D30	D31	D32	D33	PR3					
D40	D41	D42	D43	PR4					
D50	D51	D52	D53	PR5					
D60	D61	D62	D63	PR6					
D70	D71	D72	D73	PR7					
D80	D81	D82	D83	PR8					
D90	D91	D92	D93	PR9					
PC0	PC1	PC2	PC3	0					

Rys. 2. Organizacja pamięci w kartach Unique



Rys. 3. Kodowanie Manchester

z sygnałem taktującym. W środku każdego okresu występuje zbocze (rys. 3):

- narastające oznacza stan logiczny „1”,
- opadające oznacza stan logiczny „0”.

Z takiego sposobu kodowania wynika metoda synchronizacji odbiornika z nieznanym sygnałem (musi on występować cyklicznie lub rozpoczynać się ciągiem da-

nym pozwalających na przeprowadzenie synchronizacji), przy czym do jej zastosowania wymagana jest znajomość (choćby w przybliżeniu) częstotliwości (a raczej okresu) szukanego sygnału.

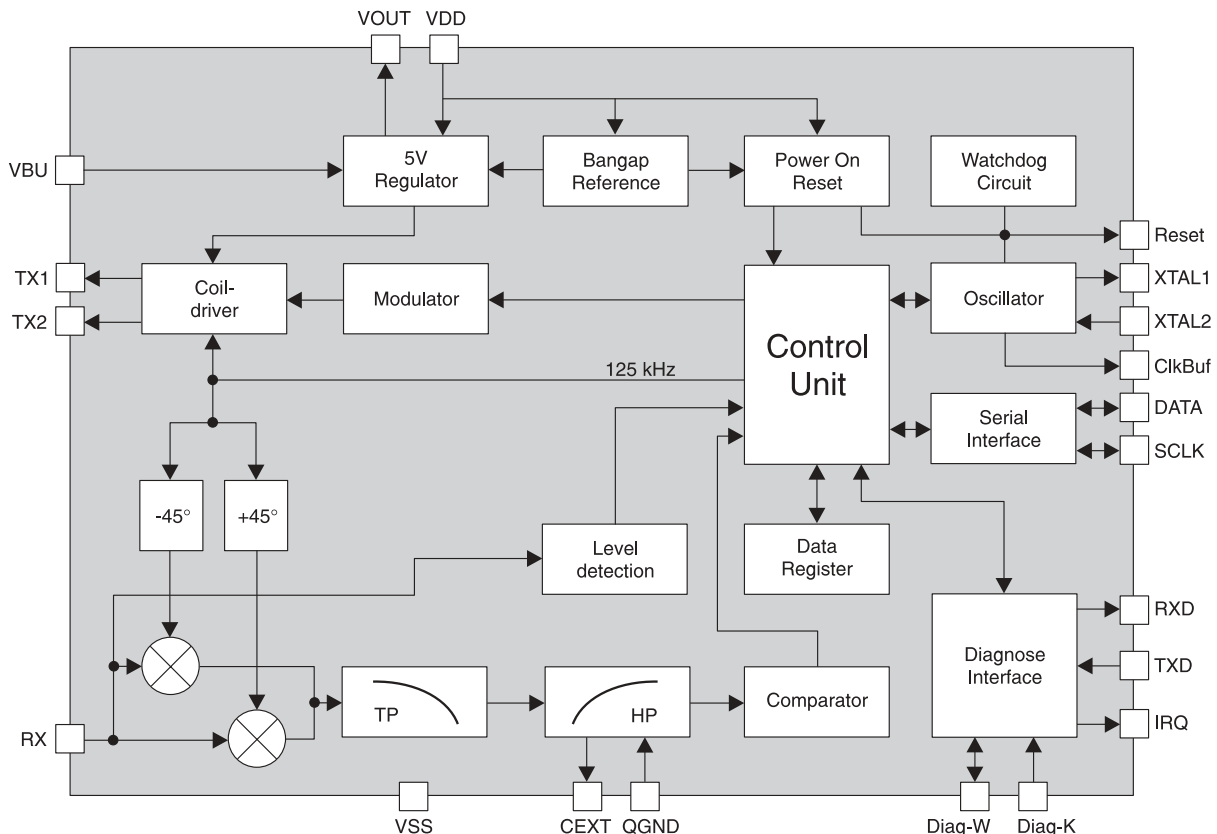
Algorytm rozpoczyna się oczekiwaniem na jakiegokolwiek zbocze. Po jego wykryciu zakładamy, że otrzymaliśmy właśnie pierwszy bit informacji - zapamiętujemy go i odczekujemy biernie czas 3/4 okresu. Następnie przez czas równy 1/2 okresu poszukujemy kolejnego zbocza. Jeśli takie wystąpi - traktujemy je jako kolejny bit strumienia danych, zapamiętujemy i rozpoczynamy kolejny cykl. Natomiast jeśli w którymś z cykli żadne zbocze się nie pojawi, będzie to oznaczało, że popełniliśmy błąd na początku, traktując krawędź bitu (początek lub koniec okresu) jako jego środek niosący informację. Wtedy niestety trzeba uznać zebrane dane za nieprawidłowe i rozpocząć całą procedurę od początku.

Metodę tę można zaimplementować w synchronicznym układzie sekwencyjnym albo - jak w tym przypadku - programowo, za pomocą mikrokontrolera.

Układ ASIC

W systemie mikroprocesorowym układ Reader Exciter ASIC stanowi interfejs między mikrokontrolerem a transponderem (w tym przypadku kartą UNIQUE). W swojej strukturze zawiera prawie cały nadawczo-odbiorczy układ w.cz. (za wyjątkiem kilku elementów pojemnościowych, dołączanych do niego), szeregowy interfejs do komunikacji z mikrokontrolerem (wraz z rejestrami danych i ustawień), układ diagnostyczny do zdalnej kontroli (nie wykorzystywany w opisywanym czytniku), generator z zewnętrznym oscylatorem kwarcowym 4 MHz (z buforowanym wyjściem, które można wykorzystać do taktowania mikrokontrolera), watchdog służący do automatycznego zatrzymywania trybu *Real-Time* (opis w dalszej części artykułu) oraz stabilizator napięcia 5 V (sam układ jest zasilany napięciem 5,3...16 V, typowo 12 V). Jego budowę i wyprowadzenia przedstawiono na rys. 4.

Komunikacja z mikrokontrolerem odbywa się za pośrednictwem dwuprzewodowego, półdupleksowego interfejsu szeregowego



Rys. 4. Schemat blokowy układu-czytnika kart bezstykowych

korzystającego z wyprowadzeń SCLK oraz DATA. Transmisja może odbywać się w dwóch trybach - normalnym oraz *Real-Time* (przebiegi czasowe komunikacji w różnych przypadkach pokazano na **rys. 5**). Po uruchomieniu układ znajduje się w pierwszym z nich, który służy do komunikacji z ASIC-iem, ustawiania parametrów pracy oraz diagnostyki. Wysłanie każdej z komend należy rozpocząć od ustawienia linii SCLK oraz DATA na poziomie niskim. Następnie należy wyzerować interfejs poprzez podniesienie do poziomu wysokiego linii SCLK, później również DATA. Pierwsze opadające zbocze SCLK oznacza początek transmisji rozkazu - kolejne jego bity (linię DATA należy ustawiać w trakcie niskiego poziomu na SCLK) są wpisywane do rejestru podczas opadania sygnału na linii SCLK.

Rozkazy o długości 8 bitów to:

- *IQ* |0|0|0|0|1|IQ-SELRQ|IQTOGRQ|x|x|;
- *Set 1* |0|0|1|0|FILTER|GAIN 1|GAIN 2|SEL 5V|;
- *Set 2* |0|0|1|1|TXEN|PD|IGN ANTFAIL|x|.

Służą one do ustawiania parametrów pracy układu. Dokładniejszy opis ich znaczenia znaleźć można w dokumentacji. Po eksperymentach okazało się, że najlepsze rezultaty dawało pozostawienie ich zgodnych z ustawieniami fabrycznymi. Dwa kolejne mają długość 4 bitów i generują 4-bitową odpowiedź. Aby ją odczytać, należy ustawić wyjście mikrokontrolera podłączonego do linii DATA w stan wysokiej impedancji, a następnie odczytywać jej kolej-

ne bity po wysłaniu zboczy narastających na SCLK:

- *Read Status* |0|1|1|0|0| zwraca czterobitowy status |ANTFAIL|SHORT|INQ|READY|, w którym poszczególne bity oznaczają kolejno uszkodzenie anteny, zwarcie w układzie diagnostycznym, aktualnie wybrany kanał demodulatora oraz gotowość układu do pracy;
- *Test Interface* |1|1|1|A|B| służy do testu transmisji szeregowej - ASIC wysyła odpowiedź zawierającą zanegowane bity A i B |0|0|0|A|B|.

Ostatnie trzy, 4-bitowe i niegenerujące odpowiedzi rozkazy wiążą się z trybem *Real-Mode*, w którym linia DATA jest przyłączona bezpośrednio do modulatora lub demodulatora w.cz., co umożliwi komunikację z transponderem (kartą), ewentualnie do wejścia wyzwalającego watchdoga, co pozwala na zrestartowanie jego wewnętrznego licznika poprzez podanie krótkiego impulsu. Są to:

- *Watchdog* |1|1|0|0|1|;
- *Data Out* |1|1|0|0|0| ustawia tryb odbioru, „podłączając“ wyprowadzenie DATA do wyjścia demodulatora;
- *Data In* |1|1|0|1|1|1| pozwala sterować modulatorem nadajnika, umożliwiając wysyłanie danych do transpondera (o ile oczywiście potrafi on je odebrać - w przypadku kart Unique tryb niewykorzystywany).

Po ustawieniu trybu *Real-Time* na linii SCLK musi być niski poziom. Wyjście z niego do trybu normalnego następuje w trzech przypadkach: po podaniu poziomu wysokiego na SCLK, po przepę-

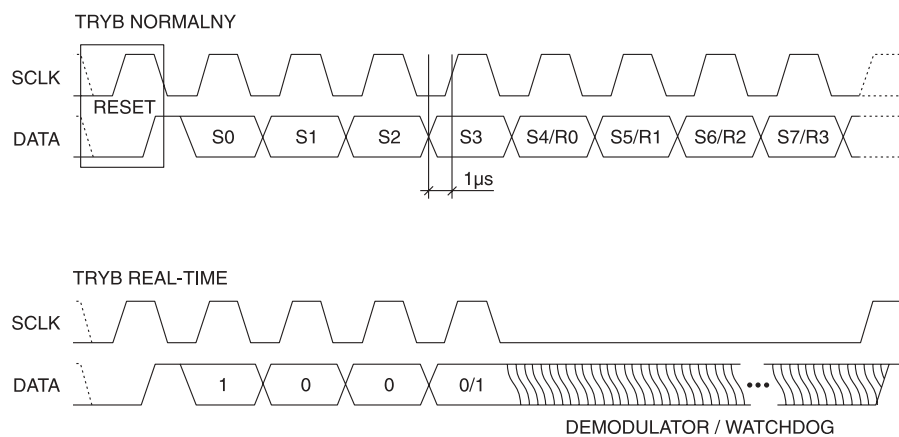
nieniu licznika Watchdoga (co następuje po 768 ms od jego wyzerowania i generuje sygnał RESET, w niniejszym czytniku ustawiający flagę zgłoszenia przerwania zewnętrznego INT0) lub po jego wyzerowaniu (poleceniem Watchdog i impulsem).

Montaż i uruchomienie

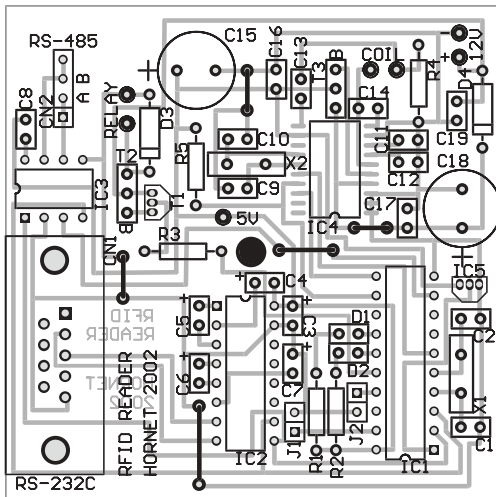
Schemat montażowy płytki pokazano na **rys. 6**. Montaż należy rozpocząć od wlutowania pięciu zwor montażowych. Następnie wlutowujemy podstawki pod układy scalone IC1...IC3 i kolejno elementy biernie - rezystory, kondensatory, diody i tranzystory oraz rezonatory kwarcowe.

Na tym etapie będziemy się posługiwać stabilizowanym zasilaczem +5 V - należy go dołączyć do oznaczonych punktów na płycie i zmierzyć napięcia zasilające na podstawkach układów scalonych i pobór prądu - większy niż kilka...kilkanaście mA oznacza zwarcie w układzie. Następnie wlutowujemy układ zerujący IC5 oraz wkładamy w podstawki wybrany układ interfejsowy i mikrokontroler z zapisanym programem testowym (*test.a51*). Jego zadaniem jest naprzemienne miganie diodami świecącymi (co 1 sekundę) i odsyłanie znaków otrzymanych przez interfejs szeregowy (ustawienia zapisane w programie to 1200 bd, 8-N-1). Wysłanie spacji powinno załączać i wyłączać zasilanie cewki przekąźnika. Należy to sprawdzić dowolnym terminalem (np. na komputerze PC). W przypadku niepowodzenia, sprawdzamy kolejno: generowanie sygnału taktującego na rezonatorze X1 (potencjalne uszkodzenie - niedokładne wlutowanie rezonatora lub kondensatorów rezonacyjnych), występowanie sygnałów transmisji szeregowej na wyprowadzeniach 2 i 3 mikrokontrolera (błąd przy montażu układu interfejsowego i ich elementów zewnętrznych lub złacz), jeszcze raz prawidłowość wlutowania wszystkich pozostałych elementów.

Po pomyślnym zakończeniu pierwszego etapu testowania można przystąpić do najtrudniejszej operacji - wlutowania układu ASIC. Ponieważ jest on wykonany w obudowie do montażu powierzchniowego, należy zachować szcze-



Rys. 5. Przebiegi czasowe ilustrujące sposób wymiany danych pomiędzy scalonym czytnikiem i mikrokontrolerem



Rys. 6. Rozmieszczenie elementów na płytce drukowanej

gólną ostrożność przy lutowaniu (zalecana lutownica grzałkowa z regulowaną temperaturą i uziemionym, cienkim grotem), zwracając uwagę na nieprzegrzewanie układu i precyzję łączenia wyprowadzeń z polami lutowniczymi.

Wyjmujemy układy z podstawek i w odpowiednie miejsce doprowadzamy +12 V (niekoniecznie stabilizowane, odłączamy przy tym wcześniej używany zasilacz 5 V). Jeśli montaż układu ASIC został przeprowadzony prawidłowo, na pinach zasilających w podstawkach pozostałych układów powinno wystąpić napięcie 5V, pochodzące z wbudowanego stabilizatora. W przypadku niepowodzenia należy przede wszystkim sprawdzić dokładnie jakość lutowania wyprowadzeń układu.

Na tym etapie trzeba wykonać ostatnią czynność montażową: wlotować antenę - cewkę o indukcyjności 1,08 mH. Sposób jej nawinięcia zależy od sposobu późniejszego montażu i żądanego zasięgu działania, ale ogólnie będzie miała kształt toroidu o średnicy kilku-kilkunastu cm, nawiniętego kilkudziesięcioma zwojami drutu DNE o średnicy 0,2 do 0,5 mm. W prototypie użyłem cewki dławika o zbliżonej indukcyjności, z którego odwinąłem kilka zwojów.

W dalszych egzemplarzach anteny nawijałem ciasno (to ważne ze względu na fakt, iż zmieniające się wymiary cewki - co następuje np. przy zginięciu zwojów podczas oklejania taśmą - zmieniają jej parametry) na walcu o średnicy 5 cm. Liczbę zwojów dobrałem

eksperymentalnie, stosując miernik indukcyjności ze zwykłego multimetru.

Następnie wkładamy ponownie układy, tym razem programując mikrokontroler programem czytnika (*rfid.a51*) i podłączamy napięcie zasilające (12 V). Pulsująca czerwona dioda i komunikat o błędnie przesłany interfejs szeregowy może oznaczać błędne wlutowanie któregoś z elementów dodatkowych układu ASIC lub nieprawidłowo dobraną cewkę. W takim przypadku należy sprawdzić sygnały z generatora kwarcowego oraz na wyprowadzeniach TX1, 2 i RX

- powinny tam występować sygnały prostokątne i piłokształtne o częstotliwości bliskiej 125 kHz.

Stale świecąca się czerwona dioda daje nadzieję na poprawne uruchomienie układu - rozwiązać wątpliwości można zbliżając kartę zgodną ze standardem Unique do anteny. Powinna zamigotać dioda zielona (sygnalizująca w tym przypadku obecność karty w zasięgu), zaś wysłanie odpowiednich komend powinno spotkać się z właściwą reakcją - odesłaniem kodu ostatnio odczytanej karty bądź „otwarcie drzwi“. Gratulacje!

Działanie programu

Program po wyzerowaniu procesora rozpoczyna działanie od ustawienia znacznika stosu oraz skoku do programu głównego, rozpoczynającego się od adresu 0100h, poza obszarem procedur obsługi przerwań. Następnie dokonuje inicjacji, wywołując podprogram INIT - ustawia w żądany sposób odpowiednie bity portów I/O, przygotowuje zawartość pamięci i flag, inicjuje zegar T0 i przerwanie zewnętrzne INTO (są wykorzystywane przez procedury komunikacji z ASIC-iem) oraz - poprzez RS_INIT - interfejs szeregowy procesora. Później dokonuje testu (błędy zgłasza podprogramem ERROR, który wysyła odpowiedni komunikat w chwili ciszy na magistrali i zawieszania wykonywanie programu, pulsująca czerwona dioda) i ewentualnie oczekuje na gotowość układu czytnika (ASIC_TEST) i przechodzi do głównej pętli programu.

WYKAZ ELEMENTÓW

Rezystory

R1, R2: 330Ω
R3: 8,2kΩ
R4: 1kΩ
R5: 33kΩ

Kondensatory

C1, C2: 33pF
C3...C7: 1μF/25V
C8, C11, C12: 100nF
C9, C10: 15pF
C13: 22nF
C14: 1,5nF
C15: 47μF/25V
C16, C17, C19: 10nF
C18: 100μF/25V

Półprzewodniki

IC1: 89C4051 (C1051, C2051) zaprogramowany
IC2: MAX232A
IC3: MAX485
IC4: Reader Exciter ASIC
IC5: DS1233
D1, D2: LED
D3: 1N4148
D4: 1N4001
T1: BC547
T2: BD135
T3: BD136

Różne

CN1: złącze D-SUB 9 pin, męskie, kątowe
CN2: złącze Gold-Pin 4 pin w 1 linii
RELAY: przekaźnik
COIL: cewka 1,08mH
X1: kwarc 12MHz
X2: kwarc 4MHz
Podstawki (20 pin, 16 pin, 8 pin)

Pętla główna jest nieskomplikowana - próbuje odczytać kartę wywołując podprogram ASIC_READ (będzie on trwał max. 768 ms - wartość układu watchdog w ASIC-u), a następnie sprawdza, czy w międzyczasie nie przyszedł z zewnątrz (przez interfejs szeregowy) rozkaz, który należy obsłużyć. Ich rozróżnieniem i wykonaniem zajmują się odpowiednie fragmenty programu, których prostota nie wymaga komentarza. Wykorzystują one podprogramy PAUSE_1S (programowe opóźnienie długości 1 s) i ANSWER (tworzy i wysyła odpowiedź na komendę zgodną ze specyfikacją protokołu). Komendy są odbierane w procedurze obsługi przerwania zgłaszanego przez układ transmisji szeregowy procesora - w pier-

wszej kolejności sprawdzany jest adres urządzenia, do którego kierowany jest komunikat (w przypadku magistrali RS485 komunikaty trafiają do wszystkich odbiorców) i jeśli pokrywa się z adresem ustawionym za pomocą zwór i odczytanym przez procedurę inicjalizacyjną, ustawiana jest flaga, która sygnalizuje, że następny znak będzie kodem rozkazu i należy go zapamiętać w odpowiedniej komórce rozkazu.

Najistotniejsze w programie są podprogramy komunikacji z układem ASIC. Procedury ASIC_SEND8, ASIC_SEND4 i ASIC_GET4 komunikują się z nim zgodnie ze specyfikacją i uzależnieniami czasowymi opisanymi w dokumentacji układu, zaś ASIC_READ przełącza go w tryb *Real-Time* i przeprowadza odbiór informacji zawartej na karcie, co przebiega następująco: program próbuje się zsynchronizować z sygnałem, oczekując na zbocze opadające (które w kodzie Manchester oznacza 0) lub narastające (1). Wpisuje uzyskaną informację do 9-bitowego bufora (składającego się z akumulatora i flagi przeniesienia) i przeskakuje do fazy oczekiwania na następny bit. Oznacza to przerwę o długości ok. 338 μ s (512 μ s, czyli czas przesyłu 1 bitu, pomnożone przez 0,66), w trakcie której może wystąpić kolejne zbocze, nieniosące jednak informacji (krawędź bitu). W tym czasie można sprawdzić, czy bufor nie został już zapełniony dziewięcioma jedynekami, które mogą oznaczać początek danych z karty (gdyby tak było, przechodzimy do następnego etapu - odczytu części informacyjnej). Po zakończeniu tej przerwy znajdujemy się gdzieś w pierwszej połowie następnego bitu, dlatego też przez 174 μ s (512...338 μ s) oczekujemy na zbocze informacyjne, którego wartość wpisujemy do bufora i przechodzimy do okresu oczekiwania na następny bit. Brak zbocza oznacza, że zsynchronizowaliśmy się w przeciwnej fazie (za pierwsze zbocze informacyjne uznaliśmy krawędź bitu), więc wracamy do etapu początkowego.

Fragment programu odczytujący część informacyjną zakłada, że synchronizacja przebiegła prawidłowo (co nie musi być prawdą,

gdyż mogła nastąpić w odwrotnej fazie na dziewięciu zerach), więc nie odmierza już okresu 174 μ s oczekiwania na zbocze informacyjne, lecz kontrolę poprawności transmisji uzależnia od bitów nieparzystości, przysyłanych co 4 bity kodu (które są kolejno umieszczane w akumulatorze, a później w tymczasowym buforze w pamięci). Niezgodność otrzymanego znacznika parzystości z obliczonym na podstawie odebranych danych sugeruje błąd synchronizacji lub transmisji i powoduje powrót do etapu początkowego.

Ostatnim etapem jest odebranie bitów parzystości kolumn, obliczenie ich na podstawie zebranych w buforze danych i ich porównanie. Niezgodność powoduje oczywiście powrót do etapu synchronizacji. Gdy porównanie wypadnie pomyślnie, następuje przepisanie informacji z bufora tymczasowego do bufora głównego i ustawienie flagi sygnalizującej obecność w nim prawidłowego kodu oraz zakończenie wykonania procedury.

Podczas trwania powyższych operacji sprawdzane jest zadziałanie układu watchdog ASIC-a, który kończy tryb *Real-Time*, sygnalizując to wysłaniem krótkiego impulsu RESET. Jego wystąpienie jest rejestrowane przez układ przerwań zewnętrznych mikrokontrolera, gdyż jest przekazywane na linię /INT0. Powoduje to natychmiastowe zakończenie podprogramu i powrót do głównej pętli.

Protokół komunikacyjny

Kontroler komunikuje się z czytnikiem za pomocą prostego protokołu, opartego o komendy zapisane znakami ASCII. Czytnik rozpoznaje dwie komendy:

- C - żądanie wysłania kodu ostatnio odczytanej karty;
- O - żądanie otwarcia drzwi - załączenie przełącznika.

Każda komenda ma postać: |ADRES URZĄDZENIA DOCELOWEGO|KOD ROZKAZU|. Tak więc, aby uzyskać kod z czytnika pracującego pod adresem 5, należy do niego wysłać (domyślnie z prędkością 1200 bd i ustawieniami 8-N-1) dwa znaki: 5C.

Odpowiedź będzie miała następującą postać: |ADRES KONTROLERA|ADRES URZĄDZENIA ODPOWIADAJĄCEGO|KOD ODPO-

WIEDZI|(ew.) DANE|. Występują następujące rodzaje odpowiedzi:

- + - polecenie wykonane poprawnie; w przypadku żądania kodu następuje po nim ciąg 10 znaków, będących zapisanym heksadecymalnie kodem karty;
- - - nie wykonano polecenia; występuje w przypadku otrzymania żądania kodu, gdy nie ma żadnego w buforze (od czasu ostatniego żądania nie odczytano żadnej karty);
- ? - nieznaną komendę - stanowi reakcję na otrzymanie komendy innej niż wyżej wymienione.

Tak więc po wysłaniu wyżej opisanego rozkazu i przy założeniu, że czytnik odczytał kartę o kodzie 0a393001243h, odpowiedź będzie następująca: 05+A393001243.

Czytnik na etapie inicjacji i testowania może sam (stosując metodę *Carrier-Sense* dostępu do medium) wygenerować komunikat o błędzie, mający format zbliżony do normalnej odpowiedzi: |ADRES KONTROLERA|ADRES URZĄDZENIA ZGŁASZAJĄCEGO|ZNACZNIK BŁĘDU|KOD BŁĘDU|

Czytnik może zgłosić dwa rodzaje błędów:

- 2 - uszkodzenie interfejsu komunikacyjnego ASIC-mikrokontroler,
- 3 - uszkodzenie anteny.

Zakończenie

Do wykorzystania mojego rozwiązania może zniechęcać nieco udziwniona metoda wydostania z czytnika kodu karty, co zostało wymuszone opisanym zastosowaniem. Zachęcam jednak do zerknięcia w kod źródłowy programu, który jest - jak mi się wydaje - bardzo dobrze skomentowany i (mimo, że assemblerowy) czytelny. Aby "namówić" czytnik do wysyłania kodu natychmiast po jego odczytaniu, wystarczy zmodyfikować dosłownie dwie linie kodu!

Na wszelkie pytania i uwagi dotyczące układu i(lub) programu czekam pod adresem hornet@sltzn.katowice.pl.

Paweł Moll

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: pcb.ep.com.pl oraz na płycie CD-EP4/2004B w katalogu PCB.