



European Coordination for Accelerator Research and Development

PUBLICATION

16 kanaowy układ sterownika silników krokowych przesuwników fazy RF 1,3 GHz zasilających nadprzewodzące wnęki rezonansowe lasera na swobodnych elektronach FLASH

Kielar, E (Niewodniczanski INSTYTUT FIZYKI JADROWEJ,
Polska Akademii Nauk, Krakow, Poland) *et al*

08 May 2011

The research leading to these results has received funding from the European Commission under the FP7 Research Infrastructures project EuCARD, grant agreement no. 227579.

This work is part of EuCARD Work Package 10: **SC RF technology for higher intensity proton accelerators and higher energy electron linacs.**

The electronic version of this EuCARD Publication is available via the EuCARD web site <<http://cern.ch/eucard>> or on the CERN Document Server at the following URL :
<<http://cdsweb.cern.ch/record/1349293>>

**INSTYTUT FIZYKI JĄDROWEJ
im. Henryka Niewodniczańskiego
Polskiej Akademii Nauk
ul. Radzikowskiego 152, 31-342 Kraków**

www.ifj.edu.pl/publ/reports/2010/

Kraków, październik 2010

Raport Nr 2044/E

**16 kanałowy układ sterownika silników krokowych przesuwników
fazy RF 1,3 GHz zasilających nadprzewodzące wnęki rezonansowe
lasera na swobodnych elektronach FLASH w ośrodku DESY,
Hamburg**

E. Kielar, K. Oliwa, W. Wierba

Abstract

The LLRF (Low Level Radio Frequency) System has to control also the RF coupling with the FLASH accelerator cavities. It is done remotely by controlling RF Phase Shifters driven by step motors. It was necessary to design the multichannel stepper motors driver controlled via Ethernet. The driver is built on the basis of low-power CMOS 8-bit AVR RISC microcontroller and the MODBUS protocol. The 16 drivers integrated on a single PCB allow for controlling 16 RF (Radio Frequency) Phase Shifters for the proper adjustment of the waveguides to cavities coupling.

1. Wstęp

Dla systemu LLRF sterującego akceleratorem FLASH zaistniała konieczność zaprojektowania sterownika silników krokowych przesuwników fazy zasilających nadprzewodzące wnęki rezonansowe. Kontroler zbudowany jest w oparciu o mikrokontrolery AVR i wykorzystuje protokół MODBUS. Zintegrowanie 16 sterowników na pojedynczej PCB umożliwia obsługę 16 silników krokowych sterujących przesuwnikami fazy RF (Radio Frequency) odpowiedzialnych za właściwe dopasowanie wnęk rezonansowych do linii przesyłowych. Poniższy opis dotyczy pojedynczego kanału sterownika – w systemie docelowym wykorzystywanych będzie 16 identycznych kanałów na jednej PCB.

2. Sterownik

Sterownikiem steruje mikrokontroler AVR RISC ATmega8 (U9) posiadający w swojej strukturze 8 kB pamięci Flash oraz 1 kB pamięci RAM. Pomiary napięcia zasilającego oraz prądu pobieranego przez silnik wymusiły zastosowanie U9 w wersji TQFP zawierającej w stosunku do wersji DIP dodatkowe 10-bitowe przetworniki A/C. W celu zapewnienia stabilności sygnału zegarowego wykorzystywanego do transmisji szeregowej oraz wyznaczania prędkości silnika mikrokontroler pracuje w konfiguracji z zewnętrznym rezonatorem kwarcowym 6MHz. Złącze P11 w standardzie Kanda umożliwia programowanie ISP mikrokontrolera.

Komunikację z hostem zapewniają układy transmisji szeregowej podłączone do portów PD0 (RXD) i PD1 (TXD) poprzez transoptory U7...U8. W sterowniku zastosowano dwa standardy interfejsu szeregowego. Wprowadzany na złącze P3 standard RS232 oparty jest o układ MAX3235 (U6), natomiast standard RS485 oparty o układ MAX485 (U4) wprowadzany jest złączem P2. Automatykę przełączania trybu pracy standardu RS485 wykonano na LMC555 (U5), pracującego w aplikacji przerzutnika monostabilnego, wyzwalanego sygnałem danych wysyłanych przez mikrokontroler. Wyboru standardu transmisji dokonuje się przełączeniem jumperów w złączu P5. Na platformie sprzętowej interfejsu szeregowego zaimplementowano protokół MODBUS RTU. Poprzez dołączenie do złącz P2 lub P3, gateway'a I-7188En-MGTCP firmy ICPCON możliwa jest transmisja w oparciu o protokół MODBUS TCP za pomocą sieci Ethernet. Schemat sterownika przedstawiony został w załączniku nr 2, natomiast schemat interfejsów szeregowych w załączniku nr 3.

Wyłączniki krańcowe podłączone są poprzez złącze P9. Lewy wyłącznik krańcowy dołączony jest do portu PD2 a prawy do portu PD3 mikrokontrolera. Rezystory podciągające R10 i R11 wymuszają prąd o wartości kilkunastu miliamperów w obwodzie czujników.

Sygnały sterujące uzwojeniami silnika z portów PD4...PD7 podawane są przez transoptory HCPL-0710 (U10...U13) do dwóch układów ULN2803A (U14...U15) zawierających w swojej strukturze po 8 tranzystorów w układzie Darlingtona o maksymalnym prądzie $I_c=0,5$ A każdy, wyposażonych w dodatkowe diody zabezpieczające. Poszczególne tranzystory połączono równolegle, uzyskując w ten sposób wyjścia o maksymalnej wydajności prądowej około 2A. Z uwagi na to, że tranzystory te są umieszczone w jednej strukturze i zapewnione są jednakowe parametry poszczególnych tranzystorów, wyjścia końcówek mocy połączone są bezpośrednio ze sobą bez wykorzystywania dodatkowym rezystorów wyrównujących prądy w poszczególnych gałęziach.

Pomiar napięcia zasilającego silnik dokonywany jest poprzez port ADC6 z wykorzystaniem dzielnika napięcia 1:20, dzięki któremu możliwy jest pomiar napięcia do wartości 50,2 V. Pomiar prądu zasilającego silnik jest realizowany z wykorzystaniem

rezystora pomiarowego (R8). Napięcie wyjściowe z bocznika jest podawane na filtr dolnoprzepustowy (R9, C21), a następnie na układ wzmacniacza nieodwracającego o wzmocnieniu ok. 10 z wykorzystaniem wzmacniacza operacyjnego LM358 (U16B). Sygnał prądowy po odfiltrowaniu w filtrze RC nadal może zawierać składową zmienną, więc z wyjścia wzmacniacza jest on następnie podawany na drugą połówkę układu LM358 (U16A), na której zrealizowano aktywny filtr dolnoprzepustowy o częstotliwości granicznej około 10 Hz. Po wyjściu z filtru sygnał jest podawany na przetwornik A/C na porcie ADC7 mikrokontrolera.

Wstępny stan sterownika można określić na podstawie sygnalizacji prezentowanej za pomocą trzech diod LED. Gdy sterownik porusza silnikiem w lewo, wówczas mruga podłączona do portu PC1 żółta LED (D5), prawe obroty silnika sygnalizuje mruganie podłączonej do portu PB2 zielonej LED (D6). Po zatrzymaniu silnika poprzez wyłączniki krańcowe wybrana dioda kierunku zapala się światłem ciągłym (kierunek prawo – zielona, kierunek lewo – żółta). Zanik sygnału z czujnika nie powoduje zgaśnięcia diody, aż do momentu wydania następnego rozkazu ruchu dla sterownika. Gdy silnik dojedzie do wybranej pozycji pośredniej zadanej z hosta lub zostanie zatrzymany sygnałem stop w pozycji pośredniej, wówczas wszystkie diody będą wyłączone (pozycja docelowa, która nie jest pozycją końcową). Do portu PC0 jest podłączona jest czerwona LED (D4), informująca o wystąpieniu w sterowniku sytuacji awaryjnej. Zaczyna mrugać w momencie, gdy napięcie zasilania nie mieści się w ustalonym przedziale lub prąd pobierany przez silnik przekroczył wartość nominalną. Ruch silnika zostaje wówczas automatycznie zatrzymany oraz zostaje odcięte od niego napięcie sterujące. Dokładną przyczynę błędów można określić poprzez odczytanie rejestru statusu A102 przez hosta.

Pełną funkcjonalność sterownik uzyskuje współpracując z hostem w oparciu o protokół MODBUS, jednak istnieje możliwość sterowania silnikiem krokowym z wykorzystaniem opcjonalnie dołączanego 3-przyciskowego panelu sterowania, co wymaga wcześniejszego skonfigurowania sterownika za pomocą hosta. Uzyskuje się wówczas możliwość sterowania kierunkiem oraz prędkością silnika, który porusza się do momentu zatrzymania lub aktywacji sygnału z czujnika krańcowego. Przycisk (S3) podłączony do portu PC2 służy do wyboru kierunku ruchu oraz zatrzymania silnika, kolejne wciskanie klawisza powoduje cykliczne obracanie się silnika w lewo, zatrzymanie silnika oraz obracanie się silnika w prawo. Naciskanie przycisku (S2) podłączonego do portu PC3 powoduje cykliczne zwiększanie prędkości silnika, aż do wartości maksymalnej, która wcześniej powinna być zdefiniowana za pomocą hosta. Naciskanie przycisku (S1) podłączonego do portu PC4 powoduje natomiast zmniejszanie prędkości silnika. Panel sterujący podłączany jest za pomocą złącza P10. Schemat panelu sterowania przedstawiony został w załączniku nr 4.

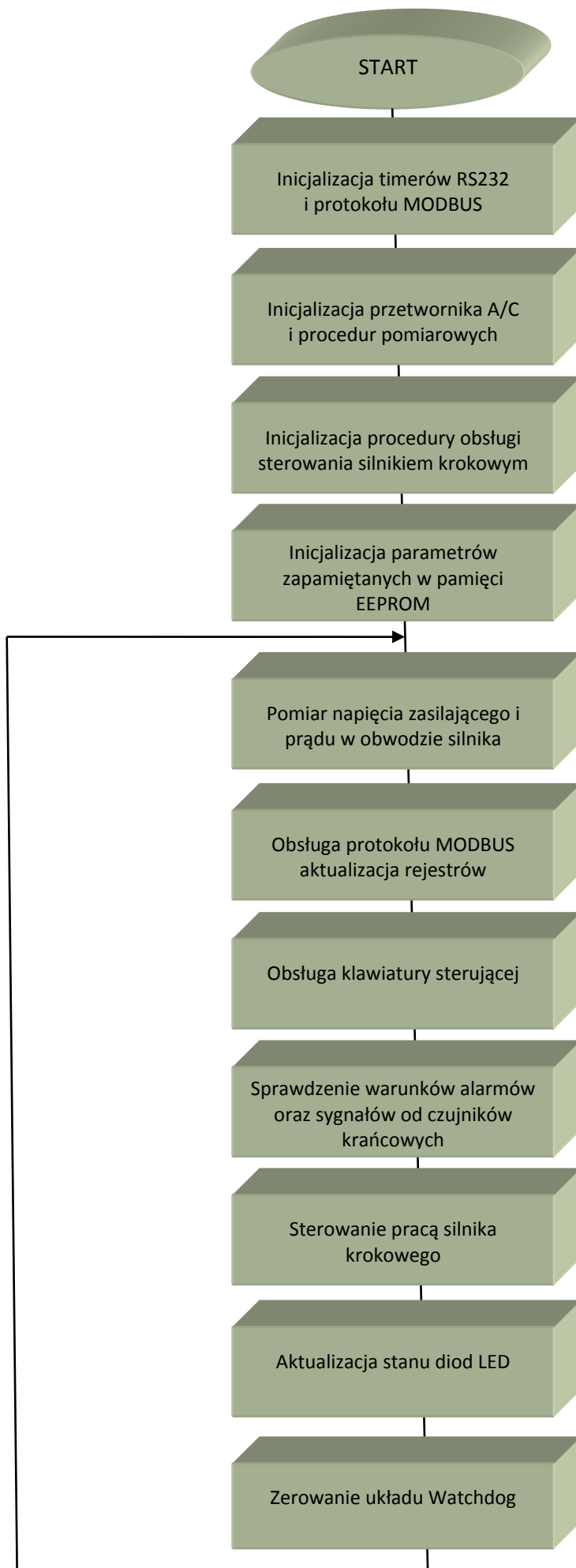
3. Układ zasilania

Napięcie zasilające V_{ss} 48V DC pochodzące z zespołu zasilaczy redundantnych podawane jest na układ przetwornicy DC/DC - V48C5C50BL (U1) na którego wyjściu otrzymujemy napięcie V_{ccD_5V} zasilające część cyfrową sterownika. Napięcie zasilające część analogową sterownika V_{ccA_5V} jest filtrowanie za pomocą filtru LC (L1, C4, C5), eliminując w ten sposób zakłócenia, jakie mogą się przedostać z silnika oraz części cyfrowej układu. Separację galwaniczną mikrokontrolera, układów interfejsów szeregowych oraz stopnia mocy zrealizowano za pomocą szybkich transoptorów HCPL-0710.

Napięcie izolowane Vcc_5V_IS do zasilania interfejsów szeregowych i transoptorów otrzymywane jest z przetwornicy DC/DC - REC3-0505SR/H1 (U2). Schemat układu zasilania przedstawiony został w załączniku nr 1.

4. Oprogramowanie

Oprogramowanie działa na postawie algorytmu z rys. 1. Po włączeniu zasilania sterownik rozpoczyna pracę od inicjalizacji interfejsu szeregowego z parametrami 9600,n,8,1; timera systemowego wykorzystywanego do odmierzania czasu oraz procedur związanych obsługą protokołu MODBUS. Ustawiany jest także przetwornik analogowo-cyfrowy, którego kanał 7 służy do pomiaru prądu pobieranego przez silnik, a kanał 8 do pomiaru napięcia zasilającego. Po tej czynności wykonywany jest kod inicjalizujący moduł sterujący pracą silnika krokowego. W efekcie uruchamiany jest m.in. 16-bitowy Timer1 sterujący pracą silnika krokowego. Następnie są ustawiane porty PD4...PD7 służące do generowania sekwencji sterujących dla silnika. Kolejną czynnością wykonywaną w procesie inicjalizacji jest odtworzenie wszystkich parametrów konfiguracyjnych z pamięci EEPROM sterownika. W pamięci nieulotnej są zapisywane następujące parametry konfiguracyjne: typ używanego silnika krokowego (2- lub 4-fazowy), tryb sterowania (pełno krokowy lub pół krokowy), sposób zachowania sterownika po zatrzymaniu silnika (włączenie lub wyłączenie prądu zasilającego silnik), aktualną pozycję w jakiej znajduje się silnik. Po odczytaniu parametrów konfiguracyjnych program wchodzi do nieskończonej pętli, w której najpierw jest odczytywana wartość z kanałów 7 i 8 przetwornika A/C, a następnie na podstawie parametrów korekcyjnych jest wyznaczane bieżące napięcie zasilające oraz prąd silnika. W następnym kroku następuje obsługa protokołu MODBUS, gdzie na podstawie danych odebranych poprzez interfejs szeregowy wykonywana jest odpowiednia akcja. Polega ona na wydaniu rozkazu dla silnika lub zapisaniu, bądź odczytaniu parametrów konfiguracyjnych oraz odesłaniu odpowiedzi zgodnie ze specyfikacją protokołu MODBUS do urządzenia nadrzędnego. Kolejną czynnością podejmowaną przez sterownik jest odczyt zmiennej zawierającej stan wciśniętych klawiszy na klawiaturze zewnętrznej i na podstawie tego zostaje zaktualizowana zmienna określająca prędkość silnika oraz jego aktualny stan (zatrzymany, kierunek lewo, kierunek prawo). Procedura odczytywania stanu klawiatury jest wykonywana tylko wtedy, gdy wcześniej za pomocą odpowiedniego rozkazu protokołu MODBUS zostało wybrane sterowanie za pomocą klawiatury. Natomiast jeżeli wybrano sterowanie pracą urządzenia za pomocą hosta, wówczas procedura nie wykonuje żadnych czynności, co uniemożliwia wydawanie sterownikowi sprzecznych rozkazów z hosta i zewnętrznej klawiatury. Następnym krokiem sterownika jest analiza sytuacji alarmowych, których wystąpienie powoduje ustawienie odpowiednich flag alarmów. W procedurze obsługi sygnałów alarmowych analizowany jest stan wyłączników krańcowych oraz napięcie zasilające i prąd silnika. W przypadku wystąpienia odpowiedniego sygnału z czujnika krańcowego lub przekroczenia wcześniej ustawionych dopuszczalnych parametrów silnika, ustawiane są odpowiednie znaczniki. Na podstawie flag alarmów i sygnałów wejściowych wysyłane są właściwe rozkazy do modułu odpowiedzialnego za bezpośrednie sterowanie silnika. Ostatnią czynnością jest sygnalizacja stanu urządzenia za pomocą diod LED. Jeżeli flaga alarmu od przekroczenia napięcia lub prądu jest ustawiona, wówczas wywoływana jest funkcja, która aktywuje miganie czerwonej diody informującej o wystąpieniu sytuacji awaryjnej. Gdy silnik obraca się w lewo wówczas uaktywniane jest błyskanie diody żółtej, natomiast w przypadku obrotu w prawo włączana jest dioda czerwona. W przypadku uaktywnienia odpowiedniej flagi oznaczającej zatrzymanie silnika na czujniku



krańcowym, zaświecana jest na stałe dioda zielona lub żółta oznaczająca kierunek ruchu. Ruch silnika następuje w wyniku podawania wektorów sterujących zawartych o odpowiednich tablicach, które następnie bezpośrednio są wysyłane do portu wyjściowego.

W sterowniku zdefiniowano po dwie tablice wektorów dla silnika 4-fazowego oraz 2-fazowego, dla trybu pełno oraz pół krokowego.

Prędkość obrotów silnika zależy od szybkości zmian wektorów sterujących, więc jest zależny od częstotliwości zgłaszania przerw, a kierunek obrotów zależy od kolejności wystawiania wektorów. Oprócz sterowania pracą silnika w procedurze przzerwiania obliczana jest także liczba kroków wykonanych przez silnik, a w przypadku, gdy silnik wykonuje ruch do określonej pozycji w momencie zrównania się licznika pozycji z wartością zadaną następuje zatrzymanie silnika w wyniku wywołania funkcji *StopMotorHw0*. Procedura ta zatrzymuje układ czasowo-licznikowy oraz w zależności od flagi określającej czy silnik ma zachować moment obrotowy, następuje wyzerowanie lub nie, linii portów wyjściowych, co powoduje wyłączenie prądu zasilającego uzwojenia silnika.

5. Rejestry MODBUS

Do komunikacji z hostem wykorzystuje się protokół MODBUS RTU. Wszystkie rejestry kontrolno-sterujące są przechowywane w ulotnej pamięci SRAM i zawartość ich jest tracona po wyłączeniu napięcia zasilającego.

Rejestry kontrolno - sterujące					
Symbol	Opis funkcji rejestru	Adres rejestru	Rozmiar rejestru w bajtach	Rodzaj operacji Read – R Write - W	Nr funkcji MODBUS
VOLTAGE	Napięcie zasilające (w 10 mV)	A100	2	R	3
CURRENT	Prąd zasilający (mA)	A101	2	R	3
STATUS	Rejestr statusu silnika	A102	2	R	3
CURR_POS	Bieżąca pozycja silnika w krokach	A103	4	R/W	3/16
TARGET_POS	Pozycja docelowa silnika w krokach	A105	4	R/W	3/16
SPEED	Rejestr prędkości silnika w krokach/s	A107	2	R/W	3/16
COMMAND	Rejestr rozkazów sterownika silnika	A108	2	R/W	3/16

Wszystkie rejestry konfiguracyjne są zapamiętywane w nieulotnej pamięci, więc zawartość ich jest przechowywana nawet po wyłączeniu napięcia zasilającego.

Rejestry konfiguracyjne					
Symbol	Opis funkcji rejestru	Adres rejestru	Rozmiar rejestru w bajtach	Rodzaj operacji Read – R Write - W	Nr funkcji MODBUS
MODBUS_ADR	Adres MODBUS sterownika	A500	2	R/W	3/16
CONFIG	Rejestr konfiguracyjny sterownika	A501	2	R/W	3/16
U_LOW_LIMIT	Dolny limit napięcia zasilającego (w 10 mV)	A502	2	R/W	3/16
U_HI_LIMIT	Górny limit napięcia zasilającego (w 10 mV)	A503	2	R/W	3/16
I_LIMIT	Limit prądu zasilającego w mA	A504	2	R/W	3/16
SPEED_LIMIT	Maksymalna dozwolona prędkość silnika	A505	2	R/W	3/16
CORRECT_U	Rejestr kontroli napięcia	A506	2	R/W	3/16
CORRECT_I	Rejestr kontroli prądu	A507	2	R/W	3/16

Zawartość rejestru STATUS (A101) umożliwia określenie stanu sterownika, znaczenie poszczególnych bitów przedstawia tabela.

Znaczenie bitów rejestru STATUS (A101)		
Nr Bitu	Symbol	Opis funkcji
0	is_run	Silnik jest uruchomiony
1	target_pos	Silnik zatrzymał się w pozycji docelowej
2	left_sig	Lewy sygnał krańcowy był aktywny
3	right_sig	Prawy sygnał krańcowy był aktywny
4	u_alarm	Napięcie zasilające poza zakresem pracy silnika
5	i_alarm	Przekroczono maksymalny dozwolony prąd silnika

Flagi alarmów zostają automatycznie wyzerowane po wydaniu nowej komendy dla silnika. Jeżeli silnik poruszał się w lewo i otrzymano alarm prawego sygnału końcowego, oznacza to uszkodzenie czujników krańcowych. Rejestr CONFIG (A501) umożliwia dostosowanie sterownika do posiadanego silnika oraz wybranie rodzaju sterowania.

Znaczenie bitów rejestru CONFIG (A501)		
Nr Bitu	Symbol	Opis funkcji
0	mot_4_phase	1 – silnik 4-fazowy 0 – silnik 2-fazowy z uzwojeniem w środku
1	mot_half_step	1 – sterowanie półkrokowe 0 – sterowanie pełno krokowe
2	kbd_control	1 – sterowanie pracą silnika za pomocą klawiatury 0 – sterowanie pracą silnika za pomocą hosta
3	nc_left_pol	1 – sygnał krańcowy lewy typu NC (Normal Closed) 0 – sygnał krańcowy lewy typu NO (Normal Open)
4	nc_right_pol	1 – sygnał krańcowy prawy typu NC (Normal Closed) 0 – sygnał krańcowy prawy typu NO (Normal Open)
5	stop_left_pwr	1 – zatrzymanie silnika zostawia zasilanie (moment obrotowy) 0 – zatrzymanie silnika nie zostawia zasilania (momentu obrotowego)

Rejestr COMMAND (A108) służy do sterowania ruchem silnika. Zapis rejestru jest możliwy gdy bit kbd_control ma wartość 0, co oznacza, że praca silnika sterowana jest przez hosta.

Znaczenie bitów rejestru COMMAND (A108)		
Komenda	Symbol	Opis funkcji
0	STOP	Rozkaz zatrzymania silnika
1	POS	Rozkaz ruchu do wyznaczonej pozycji
2	LEFT	Rozkaz ruchu w lewo do sygnału krańcowego
3	RIGHT	Rozkaz ruchu w prawo do sygnału krańcowego
4	SAVE_POS	Rozkaz zapisu pozycji silnika w pamięci EEPROM

Wydanie rozkazu dla kontrolera silnika odbywa się poprzez zapisanie rejestru COMMAND. Wydanie rozkazu jest możliwe tylko w przypadku, gdy sterownik nie wykonuje żadnego rozkazu (stan STOP). Wykonanie stanu STOP jest możliwe zawsze.

Odczyt tego rejestru umożliwia określenie stanu silnika, na przykład czy sterownik pozostaje w fazie wykonywania rozkazu. Komenda SAVE_POS służy do zapamiętania bieżącej pozycji w nieulotnej pamięci EEPROM. Pozycja ta zostaje automatycznie odtworzona po ponownym włączeniu zasilania sterownika. Pozycja silnika w jakiej się znajduje nie jest z każdą zmianą automatycznie zapisywana w pamięci EEPROM, ponieważ pamięć ta charakteryzuje się ograniczoną liczbą cykli zapisu. Zapisu pozycji w tej pamięci komendą SAVE_POS należy dokonać w momencie, gdy planowane jest wyłączenie zasilania sterownika. Po ponownym włączeniu zasilania sterownika bieżąca pozycja silnika zostanie automatycznie odtworzona.

Rejestry pozycji CURR_POS oraz TARGET_POS są 32-bitowymi rejestrami typu signed long określającymi bieżącą oraz docelową pozycję silnika. Muszą być one odczytywane i zapisywane w całości odpowiednio funkcjami 3 oraz 16. Przykładowo: wpisanie do rejestru TARGET_POS wartości -1000, podczas gdy rejestr CURR_POS zawiera

wartość 0 oraz wydanie komendy ruchu do wybranej pozycji spowoduje wykonanie przez silnik 1000 kroków w lewo. Przy pierwszym uruchomieniu sterownika należy silnik ręcznie ustawić w znanej pozycji, i tę znaną pozycję wpisać do rejestru CURR_POS, a następnie zapamiętać w nieulotnej pamięci EEPROM przed wyłączeniem zasilania sterownika poprzez wydanie komendy SAVE_POS. Pominięcie tej czynności powoduje utratę informacji o bieżącej pozycji silnika.

Rejestr SPEED określa prędkość jaką porusza się silnik. Zmiana zawartości tego rejestru podczas ruchu powoduje automatyczną zmianę prędkości silnika.

Rejestr VOLTAGE zawiera wartość bieżącego napięcia zasilającego silnik wyrażonego w jednostkach 10 mV. Na przykład, jeżeli rejestr ten zawiera 1500, to napięcie zasilające silnik ma wartość 15,00 V. Rejestry U_LOW_LIMIT oraz U_HI_LIMIT są nieulotne, umożliwiają określenie prawidłowego napięcia zasilającego silnik. Jeżeli napięcie to nie mieści się w wyznaczonym przedziale, wówczas generowana jest odpowiednia flaga alarmu (błyska czerwona dioda) oraz uruchomienie silnika nie jest możliwe. Rejestry te również wyrażają napięcie w jednostkach 10 mV. Przed pierwszym uruchomieniem sterownika należy je poprawnie skonfigurować, w zależności od nominalnego napięcia zasilania silnika.

Rejestr CURRENT wyświetla bieżącą wartość prądu pobieranego przez silnik wyrażoną w mA, natomiast rejestr I_LIMIT zawiera wartość ograniczenia prądowego, po przekroczeniu której zostaje wygenerowany alarm, a silnik zostanie natychmiast wyłączony. Przed pierwszym uruchomieniem sterownika rejestr ten należy poprawnie skonfigurować, w zależności od wartości ograniczenia prądowego.

Rejestr SPEED_LIMIT umożliwia określenie maksymalnej prędkości, z jaką może poruszać się silnik. Rejestr ten przy pierwszym uruchomieniu należy ustawić w zależności od maksymalnej dozwolonej prędkości silnika, tak aby przy ruchu z dużymi prędkościami nie gubił on swojej pozycji.

Rejestry CORRECT_U oraz CORRECT_I służą do wprowadzenia współczynnika korekcji dla toru pomiaru prądu i napięcia, z uwagi na rozrzuty produkcyjne rezystorów znajdujących się w analogowym torze pomiarowym. Przed pierwszym uruchomieniem sterownika należy przeprowadzić kalibrację wpisując odpowiednie współczynniki korekcyjne do tych rejestrów.

Współczynniki te wyznaczamy wg następującego wzoru:

$$C_u = (V_{ss}/U_w) * 1000$$

$$C_i = (I_{ss}/I_w) * 1000$$

gdzie:

C_u – zawartość wpisywana do rejestru CORRECT_U

V_{ss} – napięcie zasilające [V]

U_w – napięcie wskazywane przez sterownik w rejestrze VOLTAGE [V]

C_i – zawartość wpisywana do rejestru CORRECT_I

I_{ss} – prąd zmierzony [mA]

I_w – prąd wskazywany przez sterownik w rejestrze CURRENT [mA]

Kalibrację układu przeprowadza się jednorazowo przed pierwszym użyciem sterownika. Po dokonaniu pomiaru wartości V_{ss} należy wyliczyć współczynnik korekcji, który należy następnie wpisać do rejestru CORRECT_U. Kalibrację toru prądowego należy przeprowadzić poprzez pomiar prądu I_{ss} w obwodzie silnika podczas jego pracy, następnie należy odczytać wartość prądu wskazywaną przez sterownik oraz wyliczyć i wpisać do rejestru CORRECT_I odpowiednią wartość.

Rejestr MODBUS_ADR określa bieżący adres sterownika w protokole MODBUS. Wsady do poszczególnych silników są generowane z kolejnymi adresami MODBUS.

6. Podsumowanie

W dalszych pracach badawczych prototypu należy przeprowadzić pomiary stopnia mocy w celu rozpatrzenia ewentualności stosowania końcówki mocy w formie mostku H opartego o tranzystory MOSFET, oraz sprawdzić funkcjonalność sterownika przy transmisji poprzez sieć LAN.

7. Załączniki

1. Schemat układu zasilania
2. Schemat układu interfejsów szeregowych
3. Schemat układu sterownika
4. Schemat panelu sterującego

Acknowledgement

This work is partly supported by:

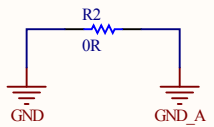
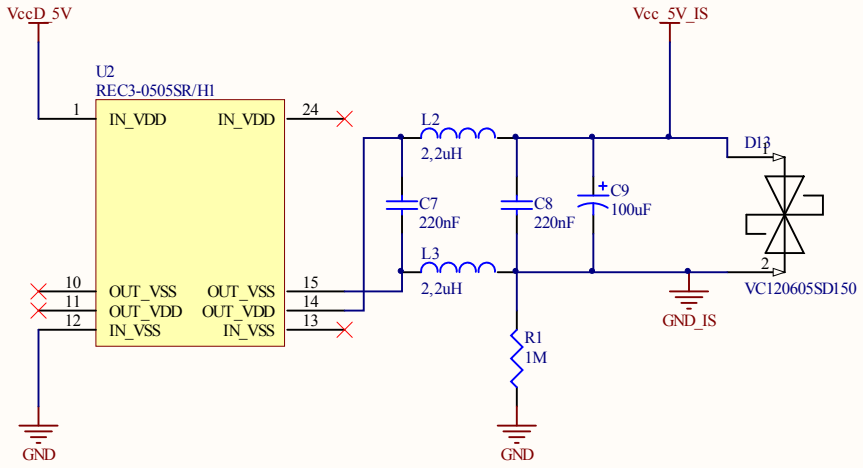
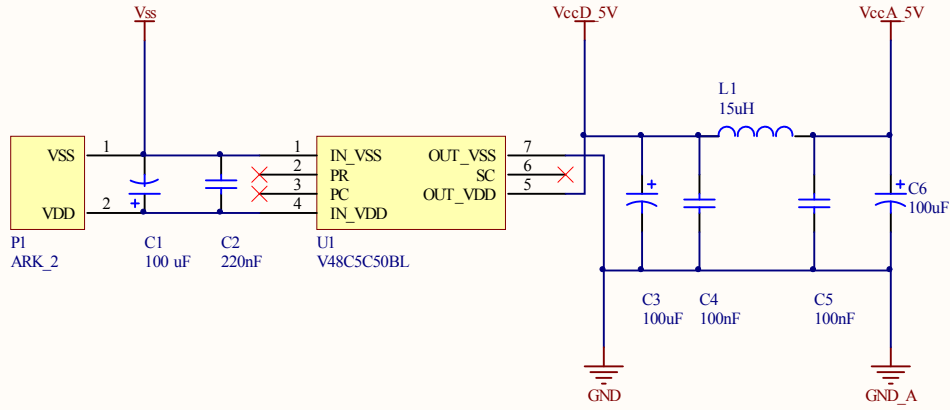
Seventh Research Framework Programme (FP7), FP7-INFRASTRUCTURES-2008-1

Akronim: EuCARD

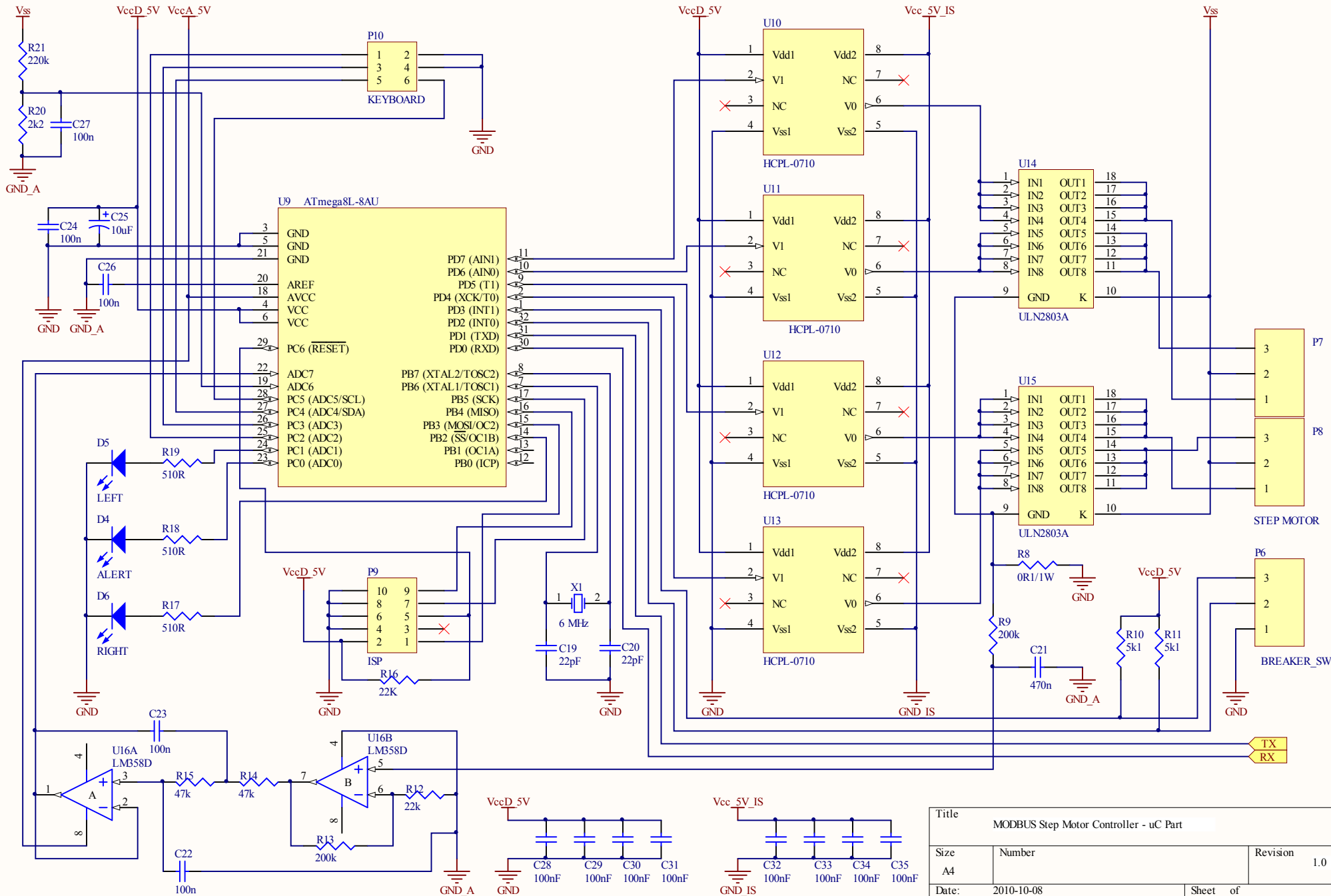
Numer kontraktu: 227579

Literatura

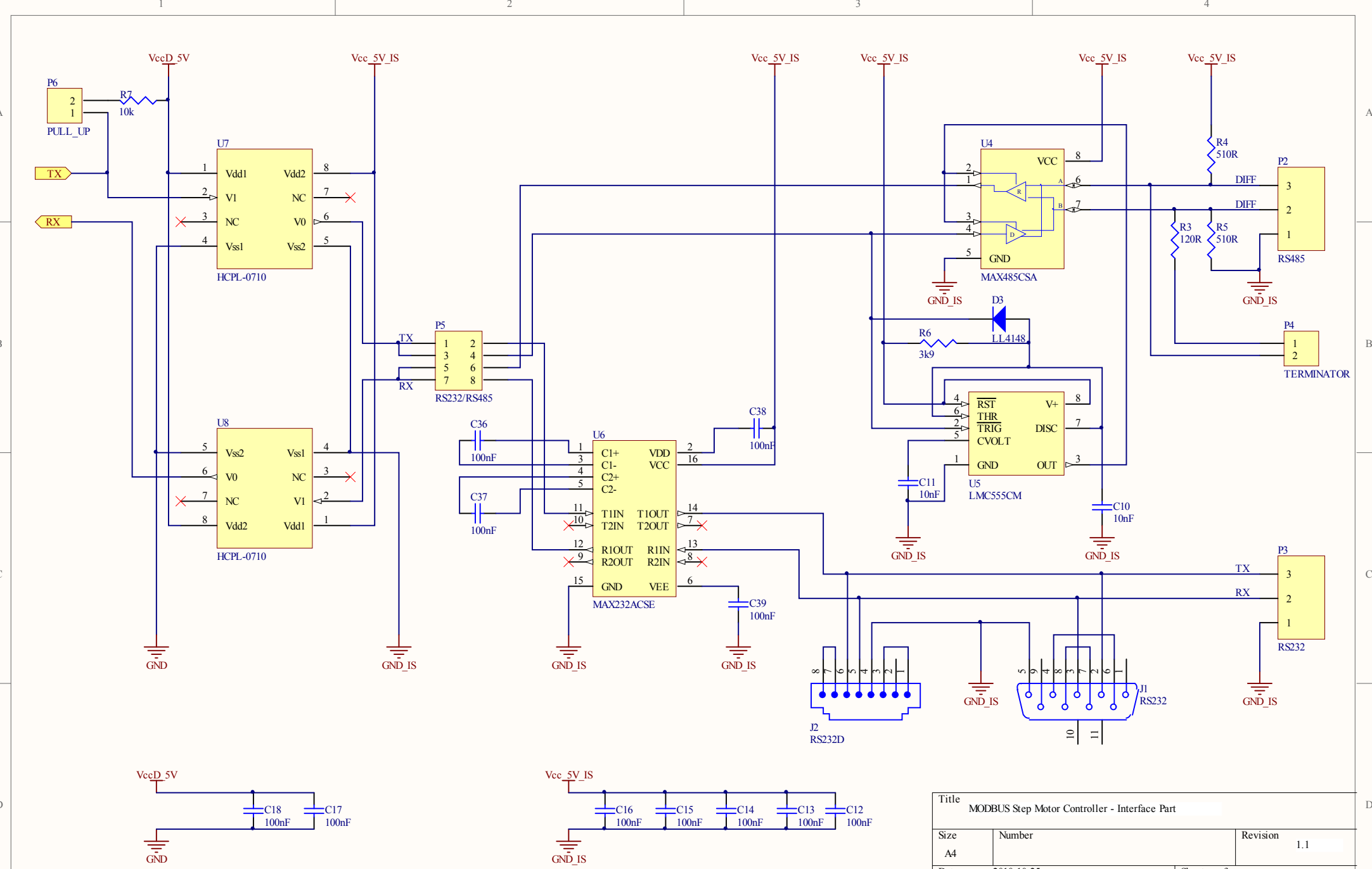
1. Atmel, Corp.: "8-bit AVR[®] with 8K Bytes In-System Programmable Flash ATmega8 ATmega8L", 2010.
http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf
2. Modicon, Inc.: „Modicon Modbus Protocol Reference Guide. PI-MBUS-300 Rev. J”, 1999. <http://www.modicon.com>
3. Modbus, Org.: „MODBUS Application Protocol Specification v.1.1.b.”, 2006
<http://www.modbus.org>
4. Mielczarek W.: „Szeregowe interfejsy cyfrowe”, Helion, Gliwice 1993.
5. Bryndza L.: „Sterownik silnika krokowego z interfejsem MODBUS”, Elektronika Praktyczna 6 i 7/2008, AVT Korporacja , Warszawa 2008



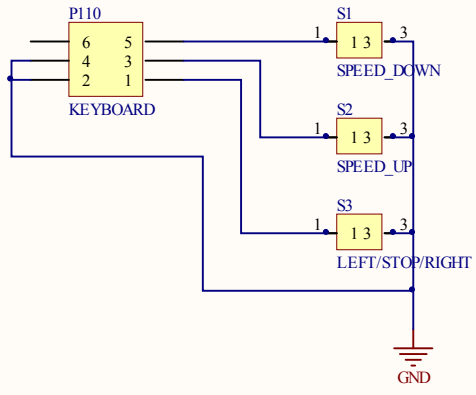
Title		
MODBUS Step Motor Controller - PSU Part		
Size	Number	Revision
A4		1.0
Date:	2010-10-04	Sheet of
File:	C:\Users\... \MODBUS Step Motor Controller - PSU Part.SchDoc	



Title		
MODBUS Step Motor Controller - uC Part		
Size	Number	Revision
A4		1.0
Date:	2010-10-08	Sheet of
File:	C:\Users\... \MODBUS Step Motor Controller	Doc By: SchDoc



Title MODBUS Step Motor Controller - Interface Part		
Size A4	Number	Revision 1.1
Date: 2010-10-25	Sheet of	
File: C:\Users\... \MODBUS Step Motor Controller Driver Interface Part v_1_1.SchDoc		



Title		
MODBUS Step Motor Controller - Keyboard Part		
Size	Number	Revision
A4		1.0
Date:	2010-10-04	Sheet of
File:	C:\Users\... \MODBUS Step Motor Controller Keyboard Part.SchDoc	