

Systemy wbudowane

Lista zadań nr 9

16 i 17 grudnia 2025

Rozwiązania należy zaprezentować najpóźniej w dniu, w którym odbywa się pracownia. Najpóźniej w tym samym dniu należy również przekazać kod źródłowy rozwiązań na SKOS. Pliki należy nazwać w czytelny sposób, podpisać w komentarzu w treści pliku, oraz przesyłać jako oddzielne pliki na SKOS – bez archiwizacji.

1. Podłącz mikrokontroler ATtiny84A¹ z zestawu elementów w następujący sposób:

nr	sygnał	połączenie
1	VCC	zasilanie (5V)
14	GND	masa (GND)
4	RESET	PB2 (D10)
7	MOSI	PB3 (D11, MOSI ATmegi328P)
8	MISO	PB4 (D12, MISO ATmegi328P)
9	SCK	PB5 (D13, SCK ATmegi328P)

Połączenia nóżek 7, 8 i 9 (MOSI, MISO i SCK) wykonaj z użyciem rezystorów $2.2k\Omega$ – takie połączenie zabezpieczy przed przepływem dużych prądów w przypadku błędów programowania obu układów. Podłącz też kondensator $100nF$ jak najbliżej nóżek 1 i 14 tego układu. Nie popełnij błędu przy podłączaniu zasilania (5V i GND) – nieprawidłowe podłączenie **uszkodzi** układ!

Pobierz ze SKOS plik `arduinoisp.hex` oraz plik `Makefile` dla ATtiny84A. Wgraj za jego pomocą program Arduino ISP na ATmega328P (`make arduinoisp`). Następnie podłącz kondensator $10\mu F$ między nóżką RST płytki Nano (ATmega328P) a masą, pamiętając o podłączeniu nóżki kondensatora opisanej znakiem minusa i paskiem (*krótszej* nóżki) do masy (minusa, linii GND). Nieprawidłowe podłączenie kondensatora **uszkodzi** kondensator! Rolą kondensatora jest zapewnienie, aby układ ATmega328P nie resetował się przy próbie użycia go jako programatora. **Uwaga:** po dodaniu tego kondensatora programowanie ATmegi328P nie będzie możliwe, aż do usunięcia kondensatora z układu.

Po wykonaniu poprzednich czynności, wgraj przykładowy program na ATtiny84A (`make install`), aby upewnić się, że wszystko działa. Program przykładowy mruga diodą, którą, wraz z rezystorem 220Ω w szeregu należy podłączyć do nóżki 5 (PB2) układu ATtiny84A.

Powtórz zadanie 1 z listy 5, wykorzystując mikrokontroler ATtiny84A.

2. Podłącz układ ATtiny84A jak w zadaniu 1. Podłącz diodę wraz z rezystorem 220Ω w szeregu do nóżki 5 (PB2) układu ATtiny84A. Natomiast do nóżki 6 (PA7) układu ATtiny84A podłącz przycisk. Podłącz też drugi przycisk do wybranej nóżki ATmegi328P.

Zaprogramuj układ ATtiny, aby działał jako SPI master, wysyłający regularnie stan naciśnięcia przycisku (np. 1 – naciśnięty, 0 – puszczony). W zależności od wartości otrzymanej od SPI slave należy też zapalać lub gasić diodę podłączoną do układu ATtiny84A. Aby przetestować program, możesz połączyć przewodem nóżki 7 i 8 układu ATtiny84A (połączenia tych nóżek z płytka Nano muszą być wykonane przy użyciu rezystorów $2.2k\Omega$ lub usunięte na czas wykonania testu – pomyłka może **uszkodzić** układ ATtiny84A lub ATmega328P!). Przy takim połączeniu naciśnięcie przycisku podłączonego do ATtiny84A powinno wywołać zapalenie diody.

Usuń teraz testowe połączenie nóżek 7 i 8 układu ATtiny84A, odłącz też kondensator $10\mu F$ od nóżki RST płytki Nano. Zaprogramuj mikrokontroler ATmega328P, aby działał jako SPI slave, z zachowaniem analogicznym do programu ATtiny84A: należy wysyłać stan naciśnięcia przycisku i sterować diodą świecącą (pin PB5) w zależności od otrzymanych danych. (Nie będzie można teraz zmieniać programu układu ATtiny84A!)

Działanie całego układu powinno być takie, że diodę podłączoną do ATtiny84A powinien zapalać przycisk podłączony do ATmegi328P, natomiast diodę podłączoną do ATmegi328P – przycisk podłączony do ATtiny84A.

¹<https://ww1.microchip.com/downloads/en/DeviceDoc/ATtiny24A-44A-84A-DataSheet-DS40002269A.pdf>

3. Podłącz układ ATtiny84A jak w zadaniu 1. Następnie zaprogramuj go tak, aby pracował jako master I²C, generując transakcje zapisu pod adres 0x7f (same jedynki), zawierające jeden bajt danych. Transakcje zapisu powinny być wykonywane regularnie (np. raz na sekundę), wartość bajtu danych powinna rosnąć o 1 w każdym kolejnym cyklu. Zastosuj bibliotekę z noty aplikacyjnej AVR310 (załączona na SKOS).

Rozłącz następnie połączenia nóżek MOSI i SCK mikrokontrolerów ATtiny84A i ATmega328P. Zamiast nich, wykonaj następujące połączenia:

nr	sygnał	podłączenie
7	SDA	PC4 (A4, SDA ATmegi328P)
9	SCL	PC5 (A5, SCL ATmegi328P)

Nie zapomnij również o rezystorach pull-up 10kΩ podłączonych do linii SDA i SCL.

Zaprogramuj mikrokontroler ATmega328P, aby pracował jako slave I²C o adresie 0x7f. Aby zaprezentować poprawność połączenia, wypisz otrzymane od mastera liczby przy użyciu UART.