

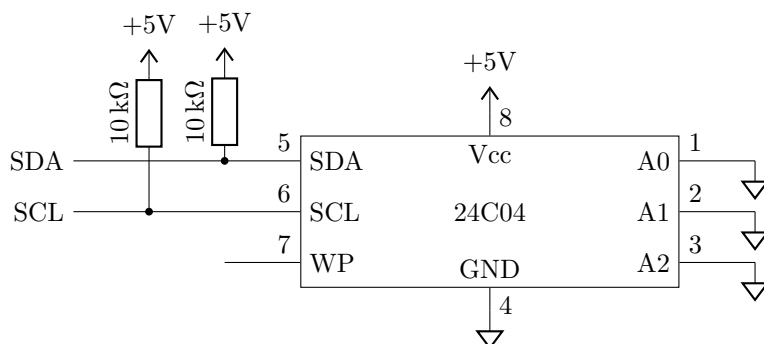
Systemy wbudowane

Lista zadań nr 7

28 listopada i 6 grudnia 2023

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 układ EEPROM I²C 24C04¹ z zestawu do pinów TWI zgodnie z poniższym schematem. Nieprawidłowe podłączenie linii zasilających **uszkodzi** układ.



Zaimplementuj program umożliwiający zapisywanie i odczytywanie danych wewnątrz układu. Możesz wykorzystać przykładowe procedury obsługi I²C z książki „Make: AVR Programming” (pliki `i2c.h` i `i2c.c` na SKOS). Komunikacja z programem ma się odbywać przez UART, program powinien obsługiwać następujące polecenia:

- `read addr` – odczyt bajtu z adresu `addr`,
 - `write addr value` – zapis pod adres `addr` wartości `value`.
2. Rozszerz program z poprzedniego zadania o następujące polecenia:
 - `read addr length` – odczyt `length` bajtów od adresu `addr`, dane należy wypisać w formacie I8HEX²,
 - `write` – zapis danych, kolejne wiersze powinny zawierać dane w formacie I8HEX.

Należy zminimalizować liczbę transakcji I²C, wykonując sekwencyjne odczyty i zapisy w ramach jednej transakcji.

3. Podłącz moduł zegara czasu rzeczywistego I²C DS3231³ z zestawu do pinów TWI, łącząc: wejście + z linią +5V, wejście - z linią GND, wejście D z linią SDA, wejście C z linią SCL. Podłącz też rezystory pull-up o wartości 10 kΩ dla linii SDA i SCL. Nieprawidłowe podłączenie linii zasilających **uszkodzi** układ DS3231. Napisz program umożliwiający ustawianie i odczytywanie daty i czasu zapisanych w układzie DS3231. Program powinien obsługiwać następujące polecenia:



- `date` – odczyt daty (w formacie DD-MM-YYYY),
 - `time` – odczyt czasu (w formacie HH:MM:SS),
 - `set date DD-MM-YYYY` – ustawienie daty,
 - `set time HH:MM:SS` – ustawienie czasu.
4. Sygnały magistrali SPI mogą być generowane programowo, przez „ręczne” sterowanie pinami GPIO (tzw. *bit-banging*). Połącz piny MISO, MOSI, SCK, SS z wybranymi czterema innymi pinami GPIO mikrokontrolera (np. PD4...PD7). Skonfiguruj wbudowany sprzętowy sterownik SPI jako slave. Napisz program, który będzie pracował jako master SPI przez *bit-banging* i komunikował się z sprzętowym sterownikiem SPI, np. wymieniając kolejne liczby 1, 2 itd. Za pomocą UART informuj o przebiegu komunikacji.

¹<https://datasheet.octopart.com/24C04WP-STMicroelectronics-datasheet-21183101.pdf>

²http://en.wikipedia.org/wiki/Intel_HEX#Named_formats

³<https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>