

Przykładowe tematy zadań inżynierskich dla przedmiotu Systemy wbudowane

Edycja 2019/2020, wersja 1.0

Zadania inżynierskie dotyczą praktycznej realizacji urządzeń wbudowanych. Działanie wykonanego urządzenia należy przedstawić osobie prowadzącej zajęcia, opracować dokumentację z realizacji zadania (sprawozdanie) oraz zaprezentować działanie wykonanego urządzenia. Preferowanym sposobem prezentacji jest kilkuminutowy film wideo (najlepiej z napisami) na temat opracowanego systemu wbudowanego.

Zadania z elementami badawczymi

Ta klasa zadań dotyczy realizacji złożonych zadań inżynierskich z wieloma aspektami badawczymi. Przy ocenianiu nacisk kładziony jest przede wszystkim na uzyskany efekt działania powstałego urządzenia wbudowanego. Dokumentacja powinna zostać opracowana pod kątem specjalistów w dziedzinie. Celem realizacji zadań jest osiągnięcie efektów kształcenia wyszczególnionych w sylabusie modułu.

1 Programator STM32 na Raspberry Pi

Wymagania technologiczne:

- platforma sprzętowa programatora: Raspberry Pi.
- docelowa platforma sprzętowa: dowolny STM32 lub Arduino

Oceniane aspekty:

- walory użytkowe,
- zaimplementowana część funkcjonalności,
- propozycja innowacyjności (potencjalnej lub zaimplementowanej),
- styl implementacji (przejrzystość kodu źródłowego, zastosowanie nowoczesnych bibliotek).

Oczekiwana funkcjonalność:

- Użyteczny i łatwy w obsłudze interfejs zewnętrzny,
- Zdalne zarządzanie konfiguracjami (własna aplikacja lub LwM2M)
- Szybkie przeprogramowywanie z wybranej, lokalnej kopii firmware'u.

Materiały pomocnicze:

- <https://learn.adafruit.com/programming-microcontrollers-using-openocd-on-raspberry-pi/overview>

2 Synchronizacja czasu - implementacja wybranego protokołu synchronizacji czasu bez aplikacyjnego systemu operacyjnego.

Wymagania technologiczne:

- platforma sprzętowa: STM32 dostępny w laboratorium (STM32F4, STM32F7),
- interfejs komunikacyjny: Ethernet,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania (optymalnie: GCC+Makefile, dopuszczalne również: STM32CubeIDE, TrueStudio).

Oceniane aspekty:

- przekonująca demonstracja działania, np. na LCD, GPIO i/lub terminalu: warto

zaprezentować działanie synchronizacji na dwóch lub więcej płytkach z jednakowym firmware'm,

- stopień uzyskania zadanej funkcjonalności
- uzyskana precyzja,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność:

- po wciśnięciu przycisku na płycie testowej następuje synchronizacja czasu przez Internet,
- czas odliczany jest w module RTC wbudowanym w mikrokontroler i tam też synchronizowany.

3 Demonstracja działania klienta LwM2M Eclipse Wakaama na urządzeniu wbudowanym bez aplikacyjnego systemu operacyjnego.

Wymagania technologiczne:

- platforma sprzętowa: STM32 dostępna w laboratorium (STM32F4, STM32F7),
- interfejs komunikacyjny: Ethernet,
- zastosowanie lwIP,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania: połączenie się przy pomocy serwera Eclipse Leshan
- demonstracja możliwości zmiany stanu i odczytu co najmniej 3 modułów czujników / układów wykonawczych.

Oczekiwana funkcjonalność:

- interakcja z modułami wykonawczymi lub czujnikami.

4 Zdalna aktualizacja firmware'u na STM32

Wymagania technologiczne:

- platforma sprzętowa: STM32 dostępna w laboratorium (STM32F4, STM32F7),
- interfejs komunikacyjny: Ethernet,
- zastosowanie lwIP,
- język programowania: C,
- narzędzia GCC i Makefile.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- walory użytkowe: czas wgrywania oprogramowania, przewidywalność,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność:

- Przez aplikację z GUI lub stronę WWW łączymy się do urządzenia, w którym chcemy uaktualnić oprogramowanie i wysyłamy nowy plik z firmware'm.

Materiały:

https://www.st.com/content/st_com/en/products/embedded-software/mcus-embedded-software/stm32-embedded-software/stm32cube-expansion-packages/x-cube-sbsfu.html

5 Implementacja szyfrowanego / bezpiecznego połączenia na platformie wbudowanej bez aplikacyjnego systemu operacyjnego.

Wymagania technologiczne:

- platforma sprzętowa: płytka testowa STM32 z interfejsem Ethernet, dostępna w laboratorium (np. STM32F4, STM32F7),
- interfejs komunikacyjny: Ethernet,
- język programowania: C,

- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania (optymalnie: GCC+Makefile, dopuszczalne również: STM32CubeIDE, TrueStudio).

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- styl implementacji (przejrzystość kodu źródłowego),
- potencjalna praktyczna przydatność opracowanego rozwiązania na podstawie propozycji zastosowań zaproponowanych w sprawozdaniu.

Materiały pomocnicze:

- <https://tls.mbed.org/>

6 Kontroler MIDI

Wymagania technologiczne:

- platforma sprzętowa: płytką testową z interfejsem USB zdolnym do pracy jako urządzenie USB,
- urządzenie powinno po dołączeniu do komputera zgłaszać się jako kontroler MIDI,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania (optymalnie: GCC+Makefile, dopuszczalne również: STM32CubeIDE, TrueStudio).

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania – współpraca z aplikacją np. wirtualnego syntezatora,
- styl implementacji (przejrzystość kodu źródłowego),
- kreatywność przy wymyślaniu interfejsu użytkownika.

Oczekiwana funkcjonalność: należy skonstruować urządzenie, które po dołączeniu do komputera zgłosi się jako kontroler MIDI. Interfejs użytkownika powinien pozwalać na generowanie wybranych dźwięków.

7 Odtwarzacz plików dźwiękowych FLAC

Wymagania technologiczne:

- platforma sprzętowa: płytką testową STM32 z wyjściem audio,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania (optymalnie: GCC+Makefile, dopuszczalne również: STM32CubeIDE, TrueStudio).

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- implementacja dekodowania FLAC,
- jakość dźwięku: brak zakłóceń i trzasków wynikających z nieprawidłowego dekodowania, płynność odtwarzania,
- styl implementacji (przejrzystość kodu źródłowego),
- walory użytkowe (interfejs użytkownika, czas reakcji).

Oczekiwana funkcjonalność: należy sprawdzić możliwość i zaimplementować w praktyce odtwarzanie dźwięku w formacie FLAC (Free Lossless Audio Codec). Można bazować na wstępnie przygotowanym projekcie odtwarzającym dźwięk w formacie Wave. Interfejs użytkownika może być minimalny (play/pause/stop).

8 Demonstracja działania jednostki MMU w mikroprocesorze ARM

Wymagania technologiczne:

- platforma sprzętowa: mikroprocesor z aplikacyjnym rdzeniem ARM Cortex-A na płycie dostępnej w laboratorium (np. Raspberry Pi lub ZYBO),
- język programowania: C,
- środowisko: darmowe; dla Raspberry Pi: GCC+Makefile, dla ZYBO: Xilinx Vivado.

Oceniane aspekty:

- przekonująca demonstracja działania,
- stopień uzyskania zadanej funkcjonalności,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność: system demonstrujący działanie jednostki MMU w nowoczesnym mikroprocesorze ARM Cortex-A.

9 Efektowne demo działania i przetwarzania danych na płycie ZYBO

Wymagania technologiczne:

- platforma sprzętowa: ZYBO,
- środowisko: Xilinx Vivado.

Oceniane aspekty:

- przekonująca demonstracja działania,
- stopień uzyskania zadanej funkcjonalności,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność: system demonstrujący działanie i przetwarzanie informacji przy pomocy platformy ZYBO.

10 Środowisko uruchomieniowe dla urządzenia wbudowanego: MicroPython

11 Środowisko uruchomieniowe dla urządzenia wbudowanego: Lua

Dla dwóch powyższych tematów:

Wymagania technologiczne:

- platforma sprzętowa o ograniczonych zasobach sprzętowych, np.:
 - jedna z płytek testowych STM32 dostępnych w laboratorium,
 - Arduino,
- działanie bez aplikacyjnego systemu operacyjnego takiego jak Unix, jednak dopuszczalny jest system operacyjny czasu rzeczywistego np. FreeRTOS.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania: wykonywanie kodu interpretowanego,
- zaimplementowana część funkcjonalności,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność:

- interakcja w języku wysokiego poziomu z układami peryferyjnymi zastosowanej platformy sprzętowej, np. GPIO, UART,
- realizacja przykładowego zadania obliczeniowego.

12 Odtwarzacz internetowych stacji radiowych

Wymagania technologiczne:

- platforma sprzętowa: płytka testowa STM32F746G-DISCO,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- styl implementacji (przejrzystość kodu źródłowego),
- walory użytkowe (interfejs użytkownika, jakość odtwarzanego dźwięku).

Oczekiwana funkcjonalność:

- odtwarzanie strumienia dźwiękowego pobieranego z przykładowej stacji radiowej.
- możliwość wprowadzania przez użytkownika adresów stacji,
- zapamiętywanie ostatnio używanej stacji oraz listy ulubionych stacji,
- regulacja głośności odtwarzanego dźwięku.

Zadania o charakterze inżynierskim

W tej klasie zadań posługujemy się nowoczesnymi, ale już popularnymi technologiami. Implementujemy przy ich pomocy taką funkcjonalność, która w wybranych aspektach prezentuje możliwości tych technologii. Tematy mają więcej aspektów inżynierskich niż badawczych. Przy ocenianiu kładziony jest nacisk na walory dydaktyczne realizacji zadania, czyli przede wszystkim:

- przejrzystość kodu i stosowanie dobrych praktyk przy pisaniu oprogramowania,
- przejrzysta dokumentacja o wysokich walorach informacyjnych, nastawiona na przekazywanie wiedzy czytelnikom również nie będącym specjalistami w dziedzinie,
- efektowna prezentacja działania.

Celem realizacji zadań jest osiągnięcie efektów kształcenia wyszczególnionych w sylabusie modułu.

1 Energooszczędny, bezprzewodowy termometr z wieloma czujnikami temperatury

Wymagania technologiczne:

- platforma sprzętowa wybrana samodzielnie,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- walory użytkowe: np. prezentacja pomiarów na płycie z mikrokontrolerem i wyświetlaczem (np. STM32F4 / F7 DISCO),
- średni pobór energii przez czujniki,
- styl implementacji (przejrzystość kodu źródłowego),
- wybór zastosowanych technologii komunikacyjnych oraz uzasadnienie zawarte w sprawozdaniu.

Oczekiwana funkcjonalność:

- Zadanie polega na skonstruowaniu termometru składającego się z wielu modułów czujnika zasilanego bateryjnie oraz modułu odbiornika, który może być zasilany z zasilacza sieciowego lub USB,
- moduły czujników powinny komunikować się z modułem odbiornika przez energooszczędny interfejs bezprzewodowy.

2 Odtwarzacz plików dźwiękowych MP3

Wymagania technologiczne:

- platforma sprzętowa: płyta testowa STM32 z wyjściem audio,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- styl implementacji (przejrzystość kodu źródłowego),
- jakość dźwięku: brak zakłóceń i trzasków wynikających z nieprawidłowego dekodowania, płynność odtwarzania,
- poprawne dekodowanie MP3,
- walory użytkowe (interfejs użytkownika).

Oczekiwana funkcjonalność: należy sprawdzić możliwość i zaimplementować w praktyce odtwarzanie dźwięku w formacie MP3 bazując na wstępnie przygotowanym projekcie odtwarzającym dźwięk w formacie Wave.

3 Eksperymentalny instrument muzyczny

Wymagania technologiczne:

- platforma sprzętowa: płytki testowa STM32 z wyjściem audio,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,
- styl implementacji (przejrzystość kodu źródłowego),
- kreatywność przy wymyślaniu interfejsu użytkownika,
- walory użytkowe, np. subiektywny czas reakcji na wciśnięcie klawisza, polifonia.

Oczekiwana funkcjonalność: należy skonstruować urządzenie, które po dołączeniu słuchawek lub wzmacniacza audio będzie działało jak elektroniczny instrument muzyczny (prosty syntezator lub keyboard z ciekawym interfejsem użytkownika).

4 Inteligentna kamera do monitoringu na Raspberry Pi wg ustalonej funkcjonalności:

Wymagania technologiczne:

- platforma sprzętowa: Raspberry Pi.

Oceniane aspekty:

- walory użytkowe,
- zaimplementowana część funkcjonalności,
- propozycja innowacyjności (potencjalnej lub zaimplementowanej),
- styl implementacji (przejrzystość kodu źródłowego, zastosowanie nowoczesnych bibliotek).

Oczekiwana funkcjonalność:

- Użyteczny i łatwy w obsłudze interfejs zewnętrzny,
- Zdalne zarządzanie konfiguracjami (własna aplikacja lub LWM2M)
- Szybkie przeprogramowywanie z wybranej, lokalnej kopii firmware'u.
- Detekcja ruchu
- Integracja z zadaniem relatywnie prostym interfejsem bazującym na porcie szeregowym (odczyt poleceń, wysyłanie statusów)
- Prosta konfiguracja, np. tekst, XML

5 Cyfrowe przetwarzanie dźwięku – demonstrator

Wymagania technologiczne:

- platforma sprzętowa: jedna z platform dostępnych w laboratorium i wyposażonych w odpowiednie interfejsy audio,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- jakość dźwięku – czy nie ma szumów, trzasków, nieprzewidzianych zniekształceń?
- stopień realizacji zadania,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność: przykłady, do wyboru:

- efekt gitarowy,
- procesor surround,
- efekty estradowe (echo, jeszcze lepiej reverb),
- wizualizacja dźwięku w czasie rzeczywistym.

6 Przykładowa sieć sensorowa z MySensors

Wymagania technologiczne:

- zastosowanie systemu MySensors.

Oceniane aspekty:

- przekonująca demonstracja działania,
- walory użytkowe,
- zawarte w sprawozdaniu kreatywne pomysły zastosowań,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność: model sieci automatyki inteligentnego budynku z wybranymi czujnikami oraz modułami wykonawczymi.

Informacje dodatkowe: <https://www.mysensors.org/>

7 Energooszczędny węzeł sieci zgodny z systemem MySensors

Wymagania technologiczne:

- Platforma sprzętowa: STM32 nucleo-32,
- zastosowanie modułu radiowego nRf24L01,
- zasilanie z pakietu baterii 2xAA ,2xAAA lub 1x ogniwo Li-Ion 18650

Oceniane aspekty:

- przekonująca demonstracja działania,
- średni pobór energii,
- walory użytkowe (subiektywnie oceniany czas reakcji przy interakcji z użytkownikiem),
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność:

- Funkcjonalność węzła sieci MySensors,
- implementacja algorytmów zarządzania energią dla zapewnienia oszczędzania baterii/akumulatora,
- implementacja przykładowej funkcjonalności ustalonej z osobą prowadzącą zajęcia: modułu wykonawczego, modułu interakcji z użytkownikiem i/lub modułu sensorycznego.

8 Obsługa modułu RFID RC522

Wymagania technologiczne:

- platforma sprzętowa: STM32 dostępny w laboratorium (STM32F4, STM32F7),
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca demonstracja działania,
- zawarte w sprawozdaniu kreatywne pomysły zastosowań,
- stopień uzyskania zadanej funkcjonalności,
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność: urządzenie demonstrujące działanie modułu RFID RC522 z mikrokontrolerem STM32 w przykładowej, użytecznej aplikacji, np. kontroli dostępu.

9 Serwer FTP

Wymagania technologiczne:

- platforma sprzętowa: STM32 dostępny w laboratorium (STM32F4, STM32F7),
- interfejs komunikacyjny: Ethernet,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca demonstracja działania,
- stopień uzyskania zadanej funkcjonalności,
- walory użytkowe (np. możliwość połączenia się przez FileZilla i upload/download plików),
- implementacja dodatkowej funkcjonalności („kreatywny dodatek”),
- styl implementacji (przejrzystość kodu źródłowego).

Oczekiwana funkcjonalność:

- urządzenie wbudowane działające jako serwer FTP przechowujący pliki na dołączonej pamięci masowej USB (pendrive),
- wybrany, kreatywny dodatek do podstawowej funkcjonalności – np. zastosowanie czujników, układów wykonawczych.

10 Implementacja systemu z brokerem i klientami MQTT

Wymagania technologiczne:

- klient: platforma sprzętowa STM32 dostępna w laboratorium (STM32F4, STM32F7),
- broker: platforma sprzętowa Raspberry Pi lub inna,
- interfejs komunikacyjny: Ethernet lub Wi-Fi,
- język programowania: C,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca demonstracja działania, najlepiej dla dwóch klientów i przesyłaniem informacji zarówno od klienta jak i do klienta,
- stopień uzyskania zadanej funkcjonalności,
- zawarte w sprawozdaniu kreatywne pomysły zastosowań.

Oczekiwana funkcjonalność:

- system urządzeń komunikujących się przez protokół MQTT.

11 Proste lub zaawansowane przetwarzanie obrazu w czasie rzeczywistym

Wymagania technologiczne:

- platforma sprzętowa Raspberry Pi z dołączonym modułem kamery,
- biblioteka OpenCV.

Oceniane aspekty:

- styl implementacji – przejrzystość kodu źródłowego (bardzo ważne),
- przekonująca demonstracja działania,
- stopień uzyskania zadanej funkcjonalności,
- zawarte w sprawozdaniu kreatywne pomysły zastosowań.

Oczekiwana funkcjonalność:

- ciekawa demonstracja na Raspberry Pi z dołączonym modułem kamery, podgląd na monitorze;
- przykłady: detektor krawędzi, detektor ruchu, śledzenie poruszających się obiektów, klasyfikacja obiektów.

12 Interaktywna aplikacja demonstracyjna dla pojazdu opartego na podwoziu Magician Chassis

Wymagania technologiczne:

- platforma sprzętowa: STM32 Nucleo64 lub Arduino UNO R3, Arduino Motor shield, ultradźwiękowy czujnik odległości, czujnik odbiciowy TCRT5000, diody WS2812
- język programowania: C, Arduino,
- darmowe (najlepiej na wolnej licencji) narzędzia do rozwoju oprogramowania.

Oceniane aspekty:

- przekonująca i efektowna demonstracja działania,

- stopień uzyskania zadanej funkcjonalności,
- styl implementacji (przejrzystość kodu źródłowego),
- możliwość rozbudowy.

Oczekiwana funkcjonalność:

- Realizacja kreatywnego pomysłu na inteligentne, autonomiczne i/lub interaktywne sterowanie pojazdem eksperymentalnym zbudowanym na bazie uniwersalnego podwozia „Magician Chassis” - szczegóły do ustalenia z osobą prowadzącą zajęcia.