

## SZKOLENIE ŚREDNIO ZAAWANSOWANE

---

# Programowanie w języku C++ dla platform ARM (seria Cortex)

CPP/ARM

Czas trwania: 4 dni (32h)

## Cele szkolenia

---

- Omówienie aspektów związanych z programowaniem na platformach mikrokontrolerów rodziny Cortex (M0-M7)
- Zapoznanie się z aspektami sprzętowymi rodziny oraz warstwą bibliotek i frameworków
- Przećwiczenie na rzeczywistym sprzęcie poruszane zagadnienia oraz nauka korzystania z fizycznych interfejsów oraz szyn komunikacyjnych

## Zalety

---

- Praktyka przed teorią - wszystkie szkolenia technologiczne prowadzone są w formie warsztatowej. Konieczna teoria jest wyjaśniana na przykładzie praktycznych zadań
- Konkretność umiejętności - w ramach każdego szkolenia rozwijamy praktyczne umiejętności związane z daną technologią i tematyką
- Nauka z praktykami - wszyscy trenerzy na co dzień pracują w projektach, gwarantuje to dostęp do eksperckiej wiedzy i praktycznego know-how

## Dla kogo?

---

- Polecamy zajęcia dla osób aktywnie programujących na platformach wbudowanych oraz znających język C i C++ w stopniu średnim

## Wymagania

---

- Znajomość języka C i C++ w stopniu średnim

## Program

---

1. Mikrokontrolery rodziny Cortex
  - a. Model architektury
  - b. Model programistyczny
  - c. Potoki i przepływ danych
  - d. Mapa pamięci
  - e. Kolejność bajtów (ang. endianness)
  - f. Operacje atomowe na bitach



- g. Timer SysTick
- h. Stany, prawa i stos
- i. System Control Block
- 2. Język asembler i właściwości platformy
  - a. Składnia języka
  - b. Instrukcje przetwarzania danych
  - c. Instrukcje sterowania przepływem
  - d. Dostęp do pamięci
  - e. Obsługa przerw
  - f. Dostęp do rejestrów specjalnych
  - g. Instrukcje koprocesora
  - h. Bariery na pamięci i synchronizacja
- 3. Procedury startowe kodu i platformy
  - a. Zadania kodu startowego
  - b. Implementacja kodu startowego
  - c. Własne wywołania i punkty wejścia
- 4. Model pamięci
  - a. Cache
  - b. Bufory zapisu
  - c. TCM
  - d. Typy pamięci
- 5. CMSIS - biblioteka kompatybilności
  - a. CMSIS-CORE
  - b. CMSIS-DSP
  - c. CMSIS-RTOS
  - d. CMSIS-SVD
  - e. CMSIS-DAP
- 6. Domena zegara
  - a. Inicjowanie domeny zegara
  - b. Funkcja powrotu do domyślnego zegara
  - c. Zegar czasu rzeczywistego i jego obsługa
- 7. Obsługa i rodzaje przerw
  - a. Model przerw
  - b. Przechwytywanie przerw
  - c. Priorytetyzacja i kontrola
  - d. Zapis do tablicy wektorów przerw i kontrolera
  - e. Wsparcie dla systemów RTOS ze strony przerw
- 8. Timery
  - a. Rodzaje i zakres stosowania Timerów
  - b. Obsługa programowa Timera
  - c. Timer systemowy SysTick
  - d. Timery i inne podsystemy mikrokontrolera
- 9. Podsystem DMA
  - a. Kanały DMA i ich obsługa



b. Kierowanie DMA do obsługi interfejsów zewnętrznych

10. Interfejsy ich obsługa

a. Port szeregowy

b. SPI

c. I2C

d. ADC

11. Kompilator i narzędzia budowania

a. Proces konsolidacji

b. Narzędzia i biblioteki systemowe

c. Zagadnienia obsługi stosu

d. Diagnostyka

