

Systemy cyfrowe i podstawy elektroniki

Adam Szmigielski

aszmigie@pjawst.edu.pl

materiały: *ftp(public) : //aszmigie/SYC*

Systemy wbudowane - wykład 15

Systemy komputerowe w sterowaniu

1. Systemy mikroprocesorowe,
2. mikrokontrolery,
3. komputery jednopłytkowe,

System Operacyjny OS (ang. *Operating System*) - definicja

- *System Operacyjny* – Oprogramowanie (zbiór programów) zarządzające zasobami komputera, tworzące środowisko do uruchamiania i kontroli zadań użytkownika (*aplikacji*),

System operacyjny - podstawowe zadania

Główne zadania systemu operacyjnego:

- Zarządzanie zasobami systemu w tym: *czasem procesora, pamięcią operacyjną,*
- Dostarcza mechanizmów do synchronizacji zadań i komunikacji pomiędzy zadaniami,
- Przydzielanie zasobów poszczególnym procesom,
- Zapewnienia równoległe wykonywanym zadaniom jednolity, wolny od interferencji dostępu do sprzętu,
- Zarządzanie bezpieczeństwem (np. dostępem do zasobów),
- Inne, np. ustalanie połączeń sieciowych, zarządzanie plikami.

Budowa systemu operacyjnego

- *Jądro systemu* - wykonuje i kontroluje zadania.

Jądro składa się z następujących elementów

- *planisty czasu procesora* - ustalającego które zadanie i jak długo będzie wykonywane,
- *przełącznika zadań* - odpowiedzialnego za przełączanie pomiędzy uruchomionymi zadaniami,
- *moduł synchronizacji i komunikacji* pomiędzy zadaniami,
- *moduł obsługi przerwania i zarządzania urządzeniami*,
- *modułu obsługi pamięci* - zapewnia przydział i ochronę pamięci,
- *inne*.

- *powłoka* – specjalny program komunikujący użytkownika z systemem operacyjnym,
- *system plików*

Podział systemów operacyjnych

Ze względu na planowanie i przydział czasu procesora poszczególnym zadaniom

- *System Operacyjny Czasu Rzeczywistego* RTOS (ang. Real Time OS),
- Systemy operacyjne czasowo niedeterministyczne.

Ze względu na sposób realizacji przełączania zadań

- Systemy z *wywłaszczaniem*,
- Systemy *bez wywłaszczania*.

Ze względu na sposób implementacji OS

- *Otwarte* systemy operacyjne,
- *Wbudowane* systemy operacyjne.

System Operacyjny Czasu Rzeczywistego

System Operacyjny Czasu Rzeczywistego (ang. Real-Time Operating System) - system operacyjny, spełniające wymogi związane z czasem wykonywanych przez niego zadań. **System zapewnia rozpoczęcie i obsługę zadań w ciągu określonego czasu.**

Zarządzanie procesami

- *Synchronizacja procesów* - w klasycznym systemie operacyjnym proces synchronizacji odbywa się z wykorzystaniem *semaforów*
- W Systemach Czasu Rzeczywistego również za pomocą *semaforów* oraz mechanizmów zintegrowanych z kolejkowaniem (np. priority inheritance protocols).

Systemy wbudowane - definicje

- Ogólna definicja *systemów wbudowanych* określa je jako urządzenia używane do kontroli, monitoringu lub wspomagania pracy urządzeń i maszyn. Pojęcie "wbudowane" odnosi się do faktu, że stanowią one integralną część systemu w którym pracują. W wielu przypadkach obecność *systemów wbudowanych* może nie być oczywista dla niewtajemniczonych obserwatorów.
- System wbudowany (ang. Embedded system) - system komputerowy specjalnego przeznaczenia, który staje się integralną częścią obsługiwanego przez niego sprzętu.

Systemy wbudowane - cd.

- System wbudowany spełnia określone wymagania, zdefiniowane do zadań które ma wykonywać,
- Typowy system wbudowany oparty jest na mikroprocesorze (lub mikrokontrolerze), zaprogramowanym do wykonywania konkretnych zadań,
- Niektóre systemy wbudowane zawierają system operacyjny.

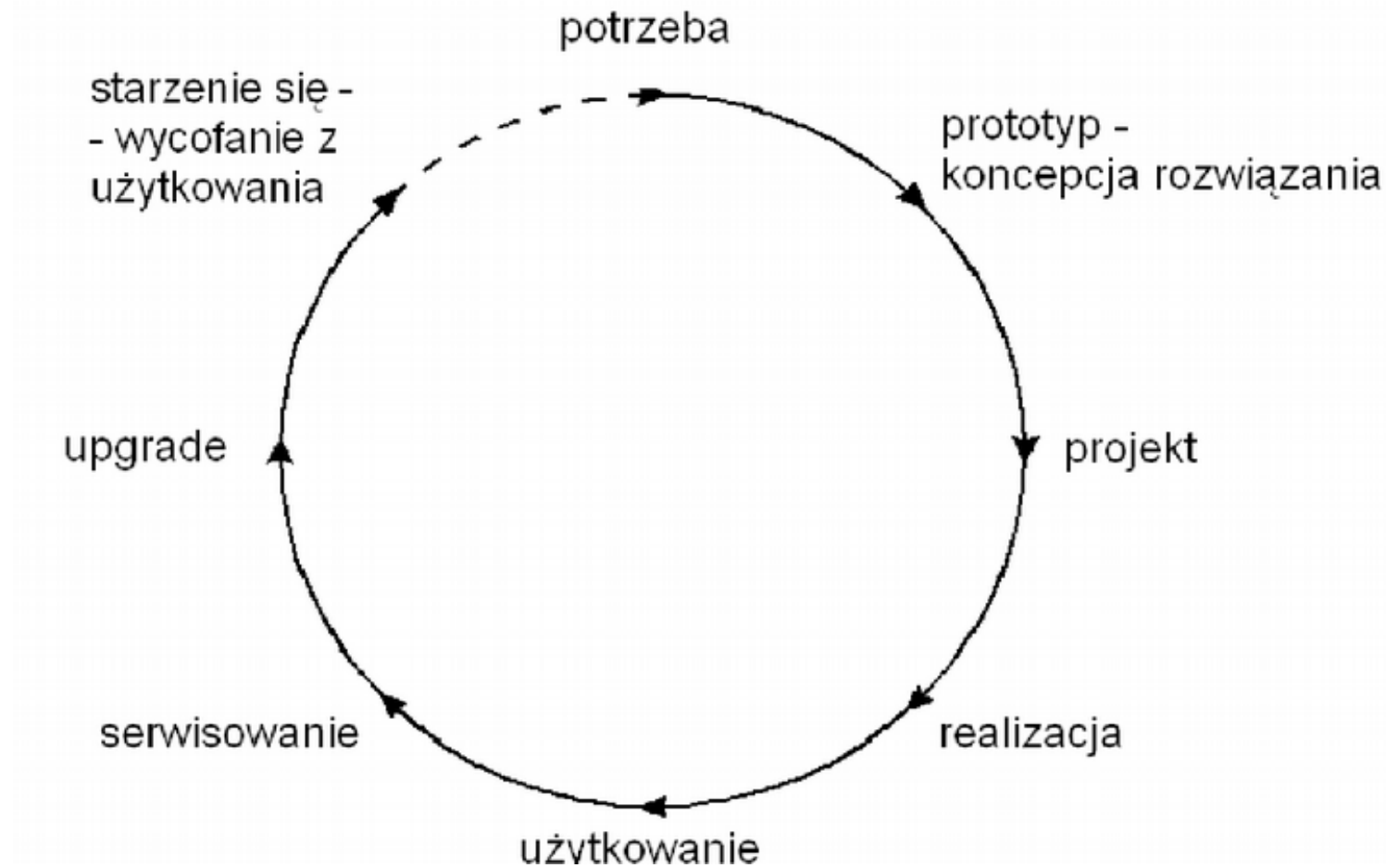
Charakterystyka systemów wbudowanych

- *Program* - zadania jakie ma wykonywać program znane są a priori,
- *Statyczne* planowanie i alokacja zasobów systemu,
- *Systemy czasu rzeczywistego* - kompromis pomiędzy użytym sprzętem i oprogramowaniem, obsługa sytuacji wyjątkowych,
- *Interakcja* pomiędzy systemem wbudowanym a zewnętrznym środowiskiem,
- *Hierarchia zachowań* - sekwencja zachowań, konkurencyjne scenariusze zachowań.

Systemy wbudowane a Komputery PC

- Prosty interfejs użytkownika (wyświetlacz, port, dioda, przyciski),
- Za pomocą portu diagnozowany może być system, w którym pracuje system wbudowany, a nie sam system wbudowany,
- Zadania programu mogą nie być bezpośrednio widoczne (program napisany pod specyficzne zastosowania).

Projektowanie systemów wbudowanych - cykl życia



Współbieżne projektowanie sprzętu i oprogramowania

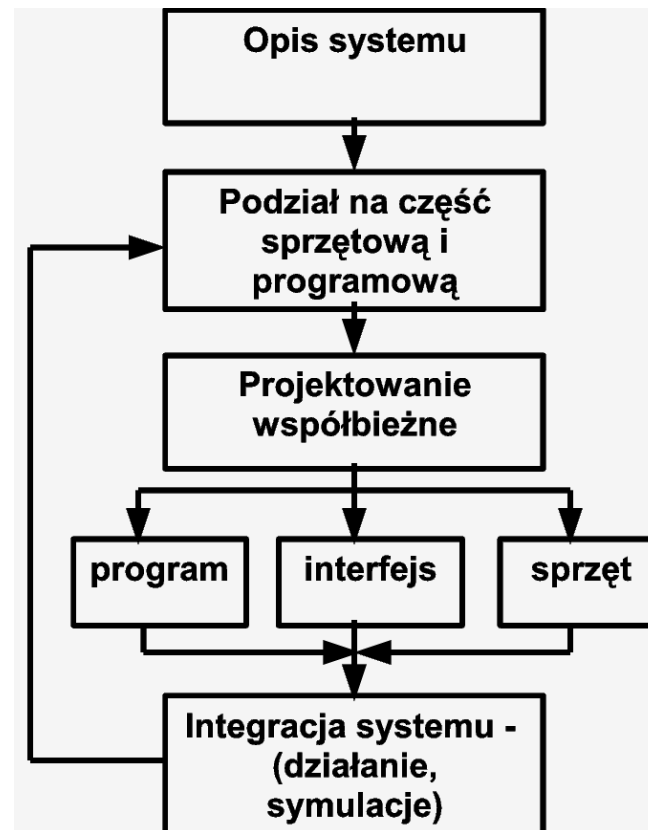


- Kooperacyjne projektowanie oprogramowania i sprzętu,
- Unifikacja osobno projektowanego oprogramowania i sprzętu,
- Wymienność pod względem funkcjonalności sprzętu i oprogramowania,

Współbieżne projektowanie sprzętu i oprogramowania - cd.

- Wspólne "wspieranie się" sprzętu i oprogramowania pod kątem stawianych celów systemowych,
- Współbieżne projektowania sprzętu i oprogramowania jest szczególnie istotne, gdy sprzęt jest jednym układem scalonym,
- W przypadku używania gotowych układów scalonych, dedykowanych do specjalnych zastosowań, cały ciężar projektowy spoczywa na części programowej,
- System powinien mieć zdolność do adaptacji do zmieniającego się środowiska, lub do niekompletnej specyfikacji.

Proces projektowania *układów wbudowanych*



- Uproszczony schemat procesu projektowania *układu wbudowanego*.

Projektowanie *systemów wbudowanych* wykorzystaniem mikrokontrolerów jednoukładowych

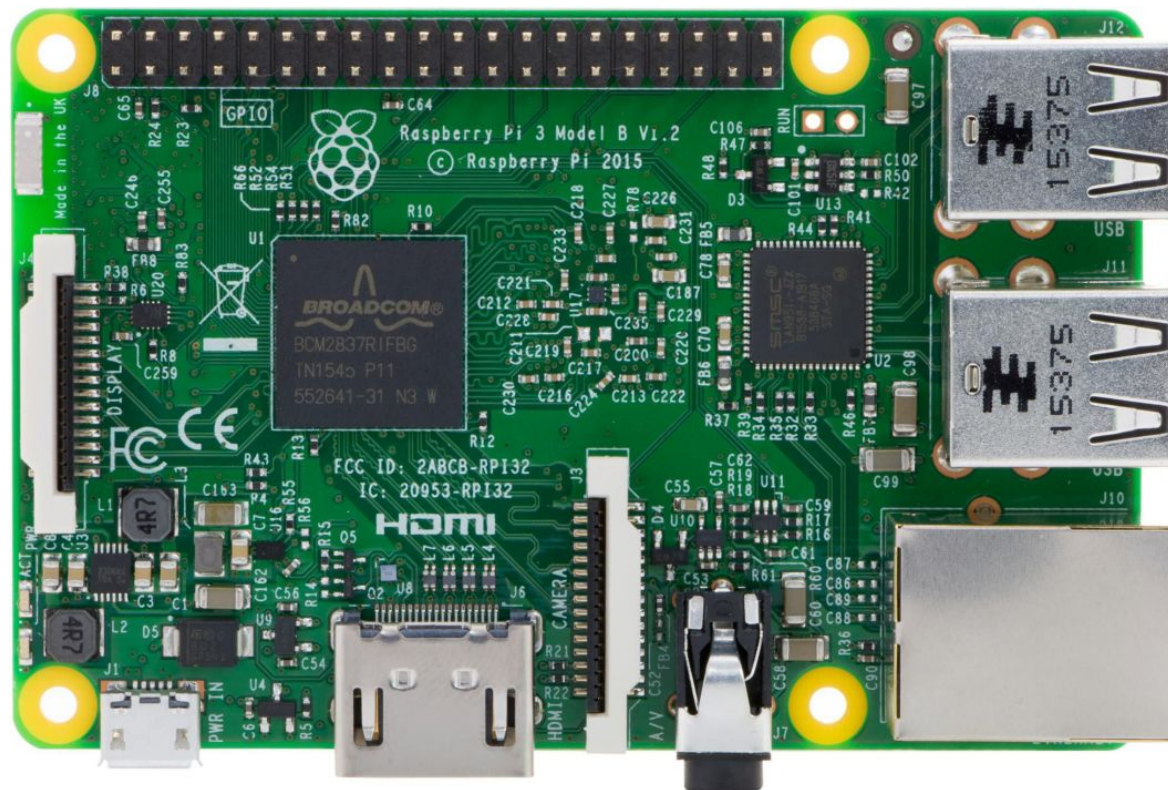
- Wybór odpowiedniego dla danego zadania mikrokontrolera ze względu na procesor, wymagane peryferia czy systemy komunikacji,
- Proces projektowania redukuje się wówczas do oprogramowania kontrolera. Pomocnym mogą być wówczas:
 - kompilatory języków wyższego rzędu,
 - dostępne biblioteki,
 - symulatory,
 - cross-kompilacja.

Charakterystyka mikroprocesorowych systemów sterowania

- Wykorzystanie zaawansowanej technologii elektronicznej – zastąpienie rozwiązań analogowych i elektromechanicznych,
- Idea stabilizującego sprzężenia zwrotnego – podstawowa zasada działania układów regulacji z wykorzystaniem systemów mikroprocesorowych,
- *Sposób pracy układów* – próbkowanie stanu procesu w dyskretnych przedziałach czasu i oddziaływaniu na proces w określonych odstępach czasu, zastosowanie logiki binarnej.
- *Projektowanie* mikroprocesorowych systemów sterowania wymaga znajomości:
 - teorii sterowania cyfrowego,

- technologii mikroprocesorowej wraz z oprogramowaniem.
- *Dokładność* – dyskretna postać sygnału odporna na szумы urządzeń pomiarowych, możliwość przesyłania na duże odległości.
- *Koszt* – rozwój technologiczny, zmniejszające się koszty wytworzenia mikrokontrolerów.
- *Nowe algorytmy* – systemy dyskretne mogą w skończonym czasie osiągnąć wartość zadaną.
- *Elastyczność* – łatwość konfiguracji regulatorów – oprogramowanie.
- *Błędy przetwarzania* – operacje: dodawania, odejmowania, błędy pomijalne w porównaniu do układów analogowych.

Raspberry Pi - komputer jednopłytkowy



- Komputer Raspberry Pi 3 Model B.

Komputer Raspberry Pi 3 Model B

- Procesor Broadcom Quad Core 1.2GHz BCM2837 64 bitów
- 1GB RAM
- BCM43143 WiFi i Bluetooth Low Energy (BLE) na płycie
- 40-pinowe rozszerzenie GPIO
- 4 x porty USB 2
- 4-biegunowe wyjście stereo i port Composite Video
- Pełnowymiarowe HDMI
- Port kamery CSI do podłączania kamery Raspberry Pi
- Port wyświetlacza DSI do wyświetlacza dotykowego
- Port Micro SD do wgrywania system operacyjnego i przechowywania danych
- Ulepszone źródło zasilania impulsowego Micro USB do 2.5A

Zadania na ćwiczenia

1. Zmontuj na płytce stykowej układ złożony z 3 diod R G B i przycisku sterującego. Zadaniem układu jest symulacja świateł drogowych.
2. Napisz program (w języku C lub Python). System powinien pracować w dwóch trybach:
 - *normalnym* Sygnał wejściowy = 1 powoduje sekwencje sygnałów świetlnych: *czzerwony* → *czzerwony i żółty* → *zielony* → *żółty*
 - *awaryjnym* Sygnał wejściowy = 0 **powoduje pulsowanie światła żółtego.**

Częstotliwość zegara zależy od trybów, tj. częstotliwość sygnału awaryjnego jest 8 razy większej od częstotliwości trybu normalnego.

3. Zrealizuj poprzednie zadanie pisząc skrypt (wykonywalny w powłoce SHELL).