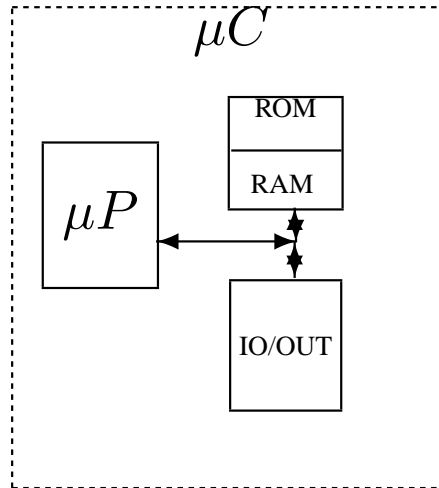


Systemy wbudowane, zastosowanie μC w układach sterownia - wykład 14

Adam Szmigielski

aszmigie@pjwstk.edu.pl

Elementy składowe mikrokontrolera μC



Podstawowe elementy składowe:

- procesor z ALU
- pamięć komputera (zawierająca *dane* i *program*)
- urządzenia wejścia/wyjścia

Mikrokontrolery

Mikrokontroler - komputer zrealizowany w postaci pojedynczego układu scalonego, zawierającego jednostkę centralną (CPU), pamięć RAM, na ogół, pamięć programu oraz rozbudowane układy wejścia-wyjścia.

Określenie mikrokontroler pochodzi od głównego obszaru zastosowań, jakim jest sterowanie urządzeniami elektronicznymi.

Typowy mikrokontroler zawiera:

- Jednostkę obliczeniową (ALU) - przeważnie 8-bitową,
- Pamięć danych (RAM),
- Pamięć programu,
- Uniwersalne porty wejścia - część tych portów może pełnić alternatywne funkcje, wybierane programowo,
- Kontrolery transmisji szeregowej lub równoległej (UART, SPI, I2C, USB, CAN, itp.),
- Przetworniki analogowo-cyfrowe lub cyfrowo-analogowe,
- timery,
- Układ kontroli poprawnej pracy (watchdog)
- wewnętrzne czujniki wielkości nieelektrycznych (np. temperatury)

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 częścią 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.

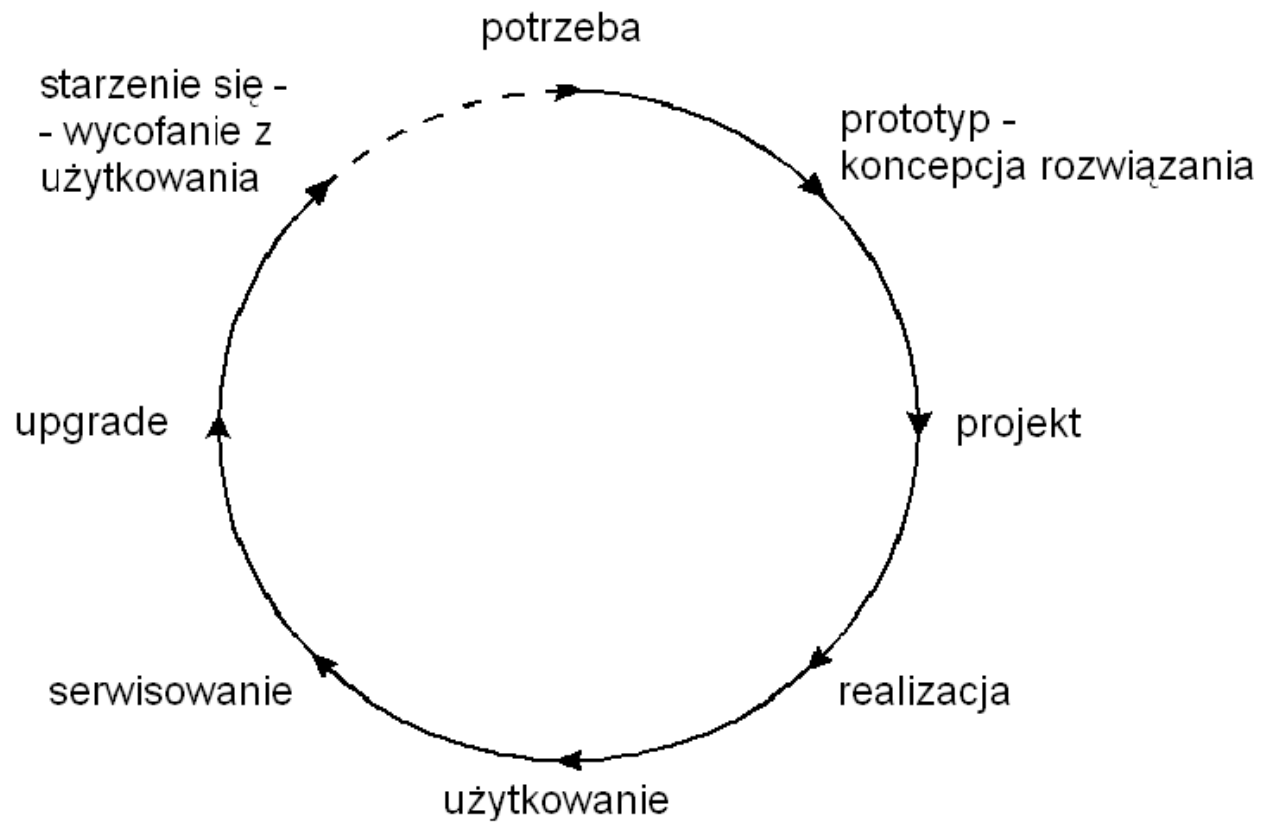
Charakterystyka systemów wbudowanych

- *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.

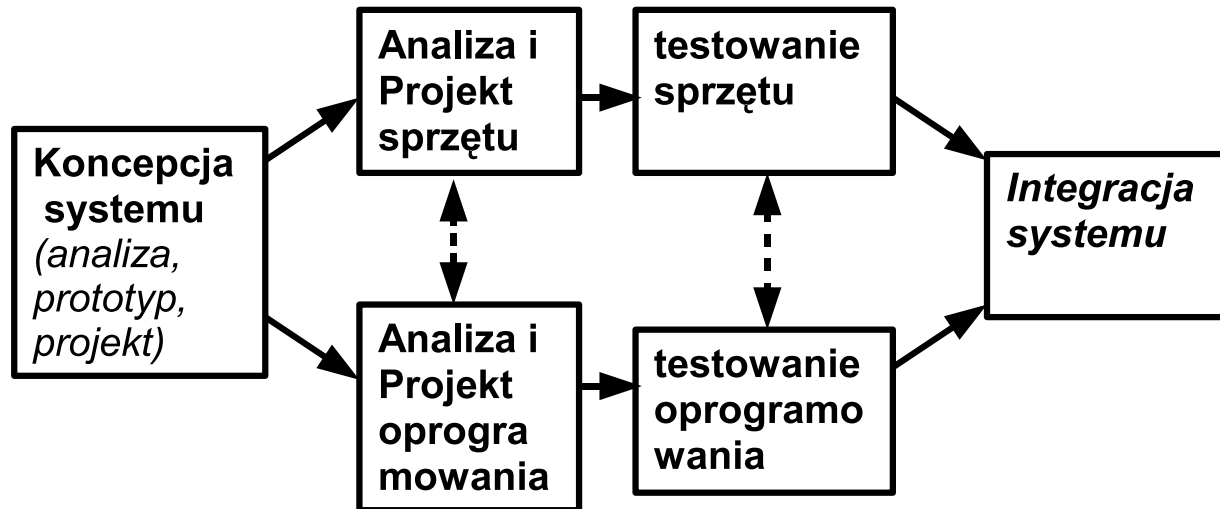
Systemy wbudowane a Komputery PC

- 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

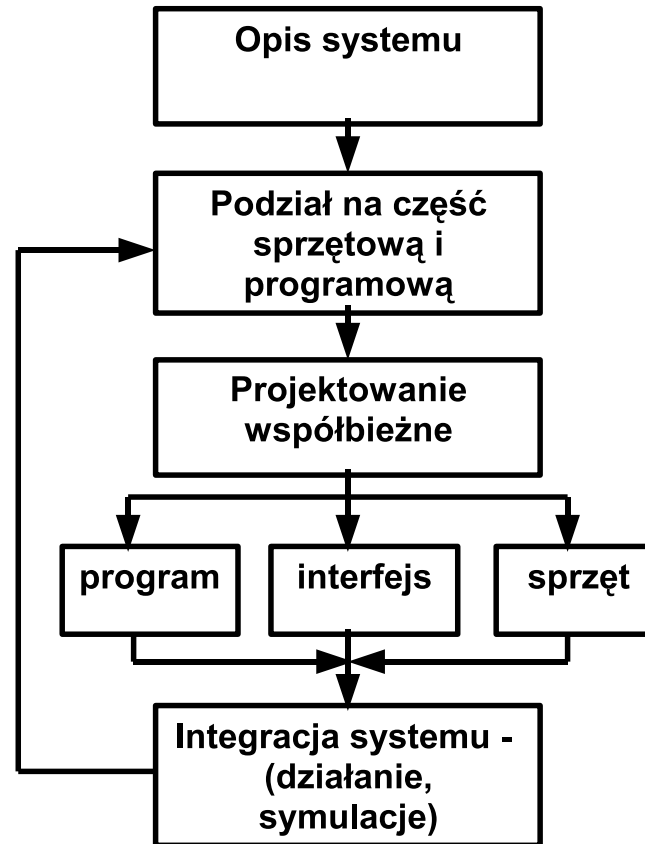


- Kooperacyjny projektowanie oprogramowania i sprzętu,
- Unifikacja osobno projektowanego oprogramowania i sprzętu,
- Wymienność pod względem funkcjonalności sprzętu i oprogramowania,

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.

Obiekt sterowania



Obiekt sterowania – obiekt, który realizuje proces (zaplanowany).

- *Fizyczny obiekt* (proces, urządzenie) jest nieodłączną częścią problemu sterowania,
- Dla projektowania sterowania niezbędna jest wiedza o fizycznym obiekcie (identyfikacja obiektu – znajomość zmiennych stanu, przepływu informacji itp.),
- *Wejście* - sygnał wejściowy, steruje naszym obiektem,
- *Wyjście* - sygnał wyjściowy, określa stan interesującej nas cechy obiektu.

Opis obiektu

- Obiekty mogą być:
 - *statyczne* - wyjście zależy tylko od wejścia (np. układy kombinacyjne, rezystor),
 - *dynamiczne* - wyjście zależy od wejścia i *stanów wewnętrznych* (układy sekwencyjne, kondensatory, cewki, sprężyny, wahadła itp.),
- Sposoby opisu obiektów:
 - *Równania stanu i wyjścia* - tak jak dla układów sekwencyjnych (automat Mealego),
 - *Transmitancja* - Laplace'a dla układów ciągłych, z-transmitancja dla układów dyskretnych,
- Czasowy opis obiektów
 - *ciągłe* - opisywane transmitancją Laplace'a albo równaniami różniczkowymi,
 - *dyskretne* - opisywane równaniami różnicowymi.

Cele sterowania

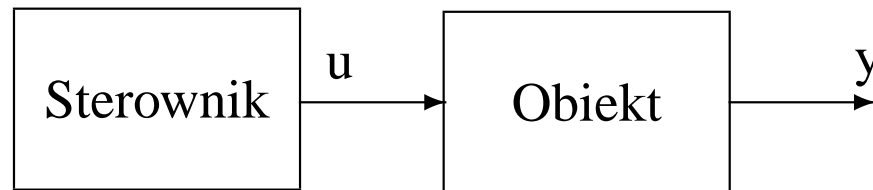
Zanim dobierzemy czujniki, elementy wykonawcze, zaprojektujemy architekturę układu regulacji musimy określić *cele* - efekty które należy osiągnąć w procesie sterowania lub po jego zakończeniu.

- *Co chcemy osiągnąć* (redukcja energii, zwiększenie zysku, ...)?
- *Jakie wielkości* należy sterować aby osiągnąć zamierzone cele?
- *Jakie są wymagania* (prędkość, dokładność, ...)?

Reguła sterowania

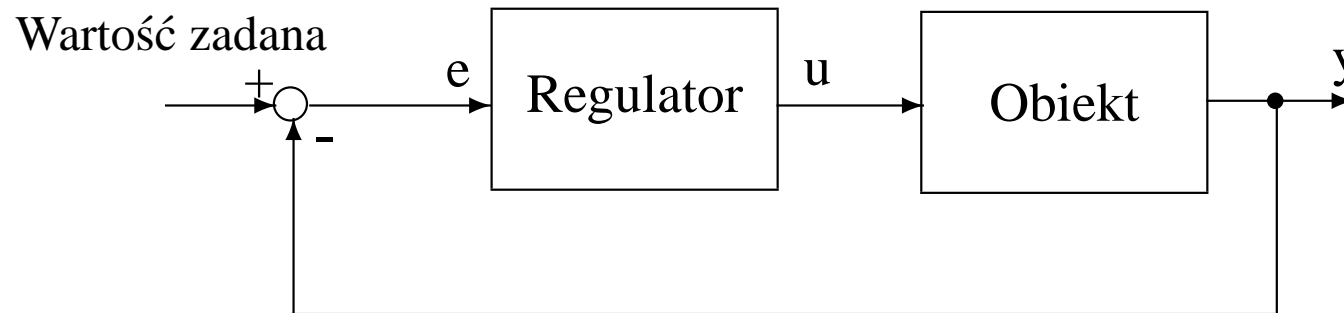
- *Zasada (algorytm)* przetwarzania informacji o stanie obiektu na sygnały sterowania elementami wykonawczymi.
- Podstawowe zasady sterowania:
 - Sterowanie w układzie otwartym,
 - Sterowanie w układzie zamkniętym ze sprzężeniem zwrotnym.

Układ z otwartą pętlą sterowania



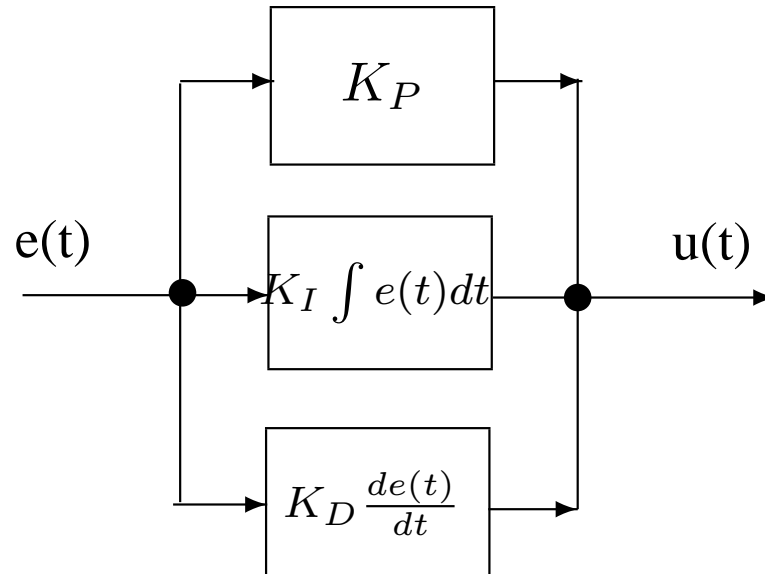
- Sterownik realizuje *cel sterowania* poprzez realizację *algorytmu sterowania*,
- Sterownik nie posiada żadnej informacji zwrotnej o przebiegu sterowania,
- Brak informacji *zwrotnej* powoduje to, że sterownik jest nieodporny na błędy sterownia.

Układ z zamkniętą pętlą sprzężenia



- Rodzaje sprzężeń:
 - *dodatnie* - wyjście wzmacnia wejście - prowadzi do destabilizacji,
 - *ujemne* - wyjście osłabia wejście - ma charakter stabilizujący,
- Układy regulacji pracują z *ujemnym sprzężeniem* zwrotnym,
- Celem układu regulacji jest osiągnięcie poprzez wyjście *y* wartości zadanej,
- To, "jak daleko" jest do osiągnięcia celu regulacji określa *błąd regulacji* - *e*,
- Regulator w zależności od błędu regulacji *e* dobiera sterowanie *u*,
- Układ ten potrafi niwelować zakłócenia.

Regulator PID (ang. *Proportional-Integral-Derivative*)



Regulator PID realizuje algorytm PID, opisany wzorem:

$$u(t) = K_P \cdot e(t) + K_I \cdot \int e(t) + K_D \cdot \frac{de(t)}{dt}$$

- Regulator składa się z członów *proporcjonalnego P*, *całkującego I* i *różniczkującego D*,
- *Nastawy regulatora* - wartości wzmocnień $\{K_P, K_I, K_D\}$.

Regulator P, PD i PI

Możliwe jest, że z regulatora usuniemy niektóre człony regulacji. W praktyce stosuje się regulatory:

- *Regulator Proporcjonalny P* - składa się tylko z członu proporcjonalnego. W układach tego typu regulacji może występować *uchyb stacjonarny*.
- *Regulator Proporcjonalno-całkujący PI* - człon całkujący powoduje eliminację *uchybu stacjonarnego*, niestety wnosi oscylacje.
- *Regulator Proporcjonalno-różniczkowy PD*

Regulator dyskretny PID

- **Regulator PID** za czasem dyskretnym reaguje w dyskretnych chwilach czasu.
- Interwał pomiędzy dwoma chwiałami T_N i T_{N-1} w których reaguje regulator nazywa się *okresem próbkowania* $t_p = T_N - T_{N-1}$.
- Regulator dyskretny PID realizuje algorytm PID w wersji dyskretnej, tj.:

$$u(nT) = K_P \cdot (nT) + K_I \cdot \sum_{k=0}^n e(kT) + K_D \cdot (e(nT) - e(n-1))$$

- Okres próbkowania powinien być możliwie mały.

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.

Charakterystyka mikroprocesorowych systemów sterowania - cd.

- *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 dyskretnie mogą w skończonym czasie osiągnąć wartość zadaną.
- *Elastyczność* – łatwość konfiguracji regulatorów – oprogramowanie.
- *Błędy przetwarzania* – operacje: dodawania, odejmowania, błędy pomijalne w porównaniu do układów analogowych.

Sterowanie - przykład

Celem sterowania jest możliwi szybki przejazd robotem zadanej odległości. Dla uproszczenia, zakładamy, że robot porusza się po prostej, brak jest przeszkód oraz elementy użyte do budowy robota są idealne.

- Sterowanie - napięcie podawane na silniki,
- Wyjście - przebyta droga proporcjonalna jest do średnicy koła i czasu trwania ruchu

Sterowanie w otwartej pętli

Mozemy tylko włączyć na zadaną chwilę silniki.

1. Musimy obliczyć jak długo powinny być włączone silniki,
2. Podać napięcie na silniki i wystartować timer na zadany czas,
3. Po zadanym czasie wyłączyć silniki.

Problemy:

1. Nieodporność na zakłócenia,
2. Bezwładność silnika. Wyłączenie napięć na silniku nie powoduje natychmiastowego zatrzymania silnika (silnik jest układem dynamicznym),
3. Bezwładność silnika może być uwzględniona w obliczeniach przed rozpoczęciem ruchu.

Sterowanie w zamkniętej pętli sterowania

1. Określić częstotliwość próbkownia - Okres próbkowania musi być co najmniej dwukrotnie mniejszy niż potrzebny czas do przebycia zadanej drogi.
2. Wybrać algorytm sterowania
3. Określić nastawy regulatora - nastawy regulatora określa się w zależności od posiadanych silników, prędkości i dokładności sterowania.
4. Podać wartość zadaną na wejście układu z zamkniętą pętlą.

Zadania na ćwiczenia

1. Zrealizuj system, złożony z serwa i czujnika odległości (alternatywnie światła), który będzie monitorował przestrzeń w poszukiwaniu najbliższej odległości (alternatywnie najmocniejszego źródła światła). Serwo powinno zatrzymać się w szukanej pozycji. A wykonanie zadania powinno zostać zasygnalizowane zaświeceniem diody LED.