

Modelul arhitectural sensor-controller-actuator

În momentul actual, nu există un standard clar definit pentru arhitectura aplicațiilor în timp real. Obiectivul unei arhitecturi de sistem standard sensor-controller-actuator pentru astfel de aplicații este obținerea unui sistem flexibil și scalabil. Extinderea sistemului, prin adăugarea de componente adiționale se realizează cu modificări minime aduse programului principal de control.

În lumea reală, datele achiziționate provin de la mai mulți senzori, în diferite momente ale rulării aplicației, iar provocările apar atât la colectarea datelor cât și la evaluarea, procesarea, luarea deciziilor, și formularea de răspunsuri adecvate care mai apoi să fie transmise către actuatori.

Deși numeroase încercări au existat în vederea realizării unei arhitecturi de sistem pentru senzori inteligenți și rețele de senzori wireless, aceste încercări nu acoperă existența actuatorilor în sistem, iar aducerea de modificări ar însemna reconfigurări considerabile a sistemelor.

În acest context, în lucrarea *“Software Architecture to Integrate Sensors and Controllers”*, autorii propun o arhitectura generică care integrează senzori, controllere și actuatori, benefică pentru o multitudine de aplicații. Arhitectura propusă utilizează o abordare bazată pe date. Datele achiziționate de la senzori sunt transmise de la un task(modul) la altul, fiecare modul procesând datele într-un anumit fel. Procesele de achiziție de date, procesare de date, comunicare și execuție de actuatori sunt efectuate separat, ceea ce conferă ușurință în adăugarea de noi funcționalități.

Arhitectura proiectată are predefinite un număr de funcționalități dorite. Pentru a avea un caracter generic, este necesar să poată fi integrate o multitudine de tipuri de senzori, care pot transmite datele pe diferite protocoale de comunicație. Datele achiziționate necesită procesare specifică fiecărui proiect în parte.

Pentru menținerea unei flexibilități ridicate este necesară separarea între procesele de achiziție de date, procesare de date și control. Acest lucru poate fi obținut prin utilizarea unor buffere de date pentru comunicarea între modulele aplicației. În momentul în care date noi sunt disponibile acestea sunt plasate în bufferele corespunzătoare, iar modulele impactate sunt notificate, eventual prin existența unor flag-uri. Anumii senzori, necesită citiri realizate ciclic. Acestea sunt implementate prin existența unui timer care la finalul perioadei de numărare indică acest lucru, urmând etapa de achiziție de date de la senzor.

Nivelurile de priorizare a task-urilor și ordinea lor de execuție în intervale de timp este predefinită în sistemul de operare utilizat pentru fiecare aplicație specifică.

În concluzie, framework-ul propus se va adresa unei serii largi de aplicații cum ar fi: monitorizare de procese, automatizări, robotica, produse automotivă. Variații ale arhitecturii propuse se întâlnesc deja în produsele din industriile menționate.