

Önálló labor összefoglaló

Szigetvári Áron

Az önálló laboratórium témái a valós idejű rendszerekkel álltak kapcsolatban.

A témalabor keretében megismerkedtem egy PC-s valós idejű operációs rendszerrel, a RT-Linuxszal, és az az által nyújtott kommunikáció időviszonyait vizsgáltam egy erre a célra készített alkalmazás segítségével.

Az önálló labor első félévében egy másik valós idejű operációs rendszert vizsgáltunk Madai Péterrel közösen, ez a QNX volt. QNX felett egy elosztott óraszinkronizációs algoritmust, az „intervallumok metszésé”-t valósítottam meg az operációs rendszer által nyújtott kommunikációs szolgáltatások segítségével.

Az önálló labor második félévében az RTI által készített NDDS nevű middleware-rel foglalkoztunk, melyet egy egyetemnek számára indított projekt keretében adtak át. Az NDDS egy publish-subscribe elven működő middleware, ami szabványos Ethernet hálózat fölé épít valós idejű szolgáltatásokat. Ennek hibatűrési lehetőségeivel azonban nem voltunk megelégedve, így arra kerestünk megoldást, hogyan lehetne azt kiterjeszteni. A rendszer ugyanis képes redundáns adatforrásokat kezelni, úgy hogy mindig a legnagyobb prioritású adatforrás jele kerül az arra feliratkozó subscriberekhez.

Az előbbieken leírt hibatűrési lehetőség természetesen csakis akkor működik jól, ha minden hibás csomópont azonnal megszünteti a kommunikációt, azaz hibás adatokat nem küld. Ez nem mindig áll fenn, szükség lehet ezért bonyolultabb szavazási vagy átlagolási algoritmusok megvalósítására. Erre alapesetben nincsen lehetőség, azonban ha bevezetünk egy csomópontot, mely megkapja több publikáció adatait, majd tetszőleges algoritmussal meghatározza a kimenő értéket, azzal megoldhatjuk a feladatot. Ez a csomópont kiadhatja az általa meghatározott értéket az adat felhasználói részére.

Ez annál is inkább megoldható, mivel lehetőség feliratkozni több publikációra egyidejűleg, wildcardk segítségével. Így a megoldás a küldő és vevő csomópontok számára is csaknem átlátszó, mindössze a küldő csomópontokban kell megváltoztatni a publication nevét pl. Temperature-ről Temperature1, Temperature2, stb-re.

Ez a megoldás felvet azonban még egy problémát, azt, hogy a „hibatűrő” csomópont meghibásodása az összes szenzor jelét elérhetetlenné teszi. Ez ellen védekezni lehet több hibatűrő csomópont létrehozásával. Sajnos ebben az esetben megmarad az a probléma, hogy mindenképpen a legnagyobb prioritású csomópont jelét fogja a rendszer továbbítani. Bizonyos esetekben azonban, például ha az adatokat kevésbé biztonságos környezetben levő szenzorok továbbítják, a hibatűrő csomópont(ok) elhelyezhetők biztonságosabb környezetben, így a módszer ebben a formájában is használható lehet.

Sajnos a szoftverhez és a dokumentációhoz is csak a félév végén sikerült hozzájutni, így a konkrét terveket tudtuk elkészíteni és megismerkedni a működéssel, kisebb próbaprogramokat futtatni a rendszeren.