

Az OPC UA-szerverek megkönnyítik a SCADA-kapcsolatok kezelését

Az adatok eljuttatása a terepi szintről a SCADA-szintig többféle protokoll szerint szervezhető. Az egyik esetben a SCADA kérdezi le az adatokat a saját ütemezőjéhez, és nem az adatok változásához időzítve; sávszélesség-vesztéséget okoz a két lekérdezési ciklus közötti idő alatt nem változó adatok többszöri átvitele, és állapotváltozások „vesznek el”, ha azok két lekérdezés között történnek. Megoldható az is, hogy a végberendezés határozza meg az adatküldés gyakoriságát – csak akkor küld adatot, ha az változik. Az OPC UA-szerverek „feltételes adatforgalmat” generáló felépítése jelentős sávszélességet takarít meg a terepi kommunikációban.

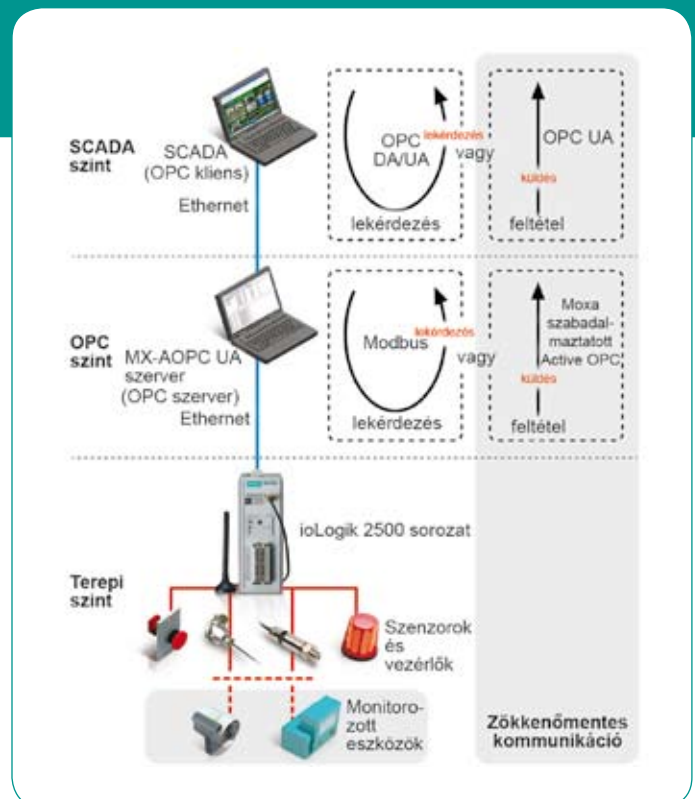
Áttekintés

Egy modern SCADA-rendszer közvetlenül csak az OPC-szerverrel van kapcsolatban, amely pedig a PLC-kkel, RTU-kkal és a Moxa távoli I/O-egységeivel kommunikál. A hagyományos OPC DA-szerver lekérdezéses módot alkalmaz, amely az adatok információ-tartalmához képest túl nagy sávszélességet használ fel. Az újabb OPC UA-szerver „feltételes adatforgalmat” generál, ezzel nagyban csökkenti a szerver és a SCADA közötti sávszélességigényt. Az OPC UA-szerver és a Moxa által szabadalmazott Active OPC-szerver kombinációja zökkenőmentes kommunikációs megoldást kínál a felhasználók számára, amelyhez lenyűgözően kis sávszélességet vesz igénybe.

Cikkünkben megvizsgáljuk a különbséget a „polling” (lekérdezés), illetve a „feltételes adatfrissítés” között, továbbá ismertetünk néhány ökölszabályt, amellyel eldönthető, melyik módszer milyen felhasználásra alkalmas. Végül, de nem utolsósorban bemutatjuk a Moxa új, MX-AOPC UA-megoldását.

Bevezetés

A SCADA-rendszerek több mint fél évszázada adják meg a lehetőséget a vezérlőszobában ülő operátorok számára, hogy eszközök és – akár nagy földrajzi területen eloszló – I/O-pontok sokaságát kövessék nyomon és ellenőrizzék. A modern SCADA-rendszer felépítését az 1. ábrán mutatjuk be, ahol a SCADA-szoftver van legfelül, a monitorozott eszközök a legalsó szinten, az OPC-szerver pedig közöttük helyezkedik el. A PLC-k, RTU-k és/vagy a Moxa ioLogik távoli I/O-egységek továbbítják a szenzorok jeleit és a mért vagy vezérlési értékeket oda-vissza az OPC-szerver és az eszközök között. Ezek az egy-



1. ábra Egy modern SCADA-rendszer struktúrája

ségek bizonyos mértékű autonómiát engednek meg a távoli helyszíneken, mert elég okosak ahhoz, hogy néhány helyi vezérlési funkciót végrehajtsanak a SCADA-szoftvertől függetlenül.

A SCADA-szoftver és az OPC-szerver hagyományosan kliens-szerver alapú lekérdezési modellt követ. Ez azt jelenti, hogy a SCADA lekérdezi az OPC-t, ami szintén lekérdezi a csatlakoztatott eszközöket



2. ábra A „subscription and monitored item” és a feltételes adatküldés

Adat-változás	Megjelenítési prioritás	
	Kritikus	Nem kritikus
Magas frekvencia	Feltétel alapú lekérdezés holtzával	Klasszikus lekérdezés (polling) magas mintavételi frekvenciával
Alacsony frekvencia	Feltétel alapú lekérdezés	Feltétel alapú lekérdezés

3. ábra A lekérdezés kiválasztása

az aktuális értékekről, és az operátorok ezen információk birtokában tehetik meg a szükséges lépéseket. Azokat a szenzorokat, amelyek kritikus helyen vannak (pl. hogy egy ajtó nyitva vagy csukva van-e) akár másodpercenként is le kell kérdezni, hogy pl. az esetleges intézkedéseket meg lehessen kezdeni. Például, ha egy ajtó állapotát 5 másodpercenként kérdezzük le, de az ajtó csak egy 4 másodperces időtartamra volt nyitva, a SCADA-rendszer nem is értesül az állapotváltozásról. Ha csak egyetlen ajtót kell figyelni, akkor a gyakori lekérdezés nem lehet probléma, de ha ajtók százait kell monitorozni, akkor ez a módszer nem használható, mert teljes sávzsélességet felhasználó és lassú alkalmazást eredményezne.

Körülbelül tíz évvel ezelőtt mutatta be a Moxa a szabadalmaztatott „Active OPC”-konceptiót, amelyet implementált is az ioLogik termékekben. Ennek az a lényege, hogy az egyszerű I/O-eszközökbe intelligenciát csempész, hogy ezek kezdeményezzék a kapcsolatot az OPC-szerverrel. Más szóval, a csatlakoztatott I/O-eszközöket az ioLogik képes periodikusan lekérdezni, de csak az előre beállított paramétereknek megfelelő adatot küldi el az Ethernet-hálózaton az OPC-szerver számára. Ahogyan az 1. ábrán is látható, a hagyományos kliensszerver polling modell a periodikus lekérdezésen, míg az „Active OPC” push-modellje az adat feltételes elküldésén alapul.

2008-ban, új fejlesztésként, az OPC Foundation szabványosította a feltétel alapú adatküldési módot (report by exception), amelyet OPC Unified Architecture-nek (röviden OPC UA-nak) nevezett el. Az OPC UA „feliratkozás és monitorozás” (subscription and monitored item) elnevezésű modellt használ a SCADA- és az OPC-szerver közötti kommunikációra. Ez a teljesen új struktúra lehetővé teszi, hogy a SCADA-rendszeren keresztül közvetlenül konfigurálható legyen az OPC-szerver és az I/O-eszközök kapcsolata. Ezzel a fejlesztéssel olyan rendszer valósítható meg, ahol a terepi eszközök is és a SCADA-szoftver is csak akkor küldenek és kapnak adatot, ha az új információt hordoz, amivel töredékére csökkenthető a felhasznált sávzsélesség.

Adatok frissítése feltétel vagy polling alapon

Sok évig a polling-alapú lekérdezés volt a szabványos ipari kommunikáció az OPC-szerver és a kliensek (értsd SCADA) között. Ma a mérnököknek már megvan a lehetőségük, hogy válasszanak a hagyományos polling és a feltétel alapú lekérdezés közül. Általában két tényező határozza meg, hogy melyik módszert érdemes alkalmazni: (1) az adat változásának a gyakorisága, valamint (2) milyen

sürgősen kell értesülnünk az állapotváltozásról. Azok a szenzorok, ahol az értékek sűrűn változnak, sűrűbb mintavételre szorulnak a pontos nyomon követhetőség érdekében. A nem gyakran változó értékeket nem szükségszerű gyakran kiolvasni, de ha túl ritkán olvasunk, előfordulhat, hogy egy kritikus esemény kimarad.

Nézzük meg részletesen, hogyan működik mindez egy OPC UA-szerveren.

Lekérdezéses adatfrissítés

Az összes OPC UA-szerver továbbra is támogatja a lekérdezéses adatfrissítést, a működési mód ez esetben azonos a hagyományos OPC DA szerverekével.

Feltétel alapú adatfrissítés

A feltétel alapú adatküldési mód esetén az OPC UA-szerver „subscription and monitored item” módszert alkalmaz, amelyben a SCADA kliens

feliratkozik a figyelt elemek listájára. A szerver egyenletes időközönként mintavételezi az értékeket, sorba állítja őket, majd állandó időközönként közzéteszi ezeket. Ha az érték nem változott az utolsó kiolvasás óta, nem kerül be a kiküldési sorba, és nem lesz közzétéve. Ez a feltétel alapú frissítés lényege (2. ábra). Megjegyezzük, hogy az OPC UA támogatja az életjelek küldését is, azaz a hosszú inaktív időszakokban is jelzi a kapcsolatban résztvevőknek, hogy a link még aktív, nem kell lezárni.

Ehhez a lekérdezési formához az OPC-kliensen két beállítás szükséges: a mintavételezési, illetve az adatküldési időközök. A mintavételi időköz határozza meg, hogy a szerver milyen sűrűn vegyen mintát az adott jelből, míg az adatküldési időt ahhoz kell igazítanunk, hogy milyen sűrűn akarunk értesülni a változásokról. A mintavételi intervallum lehet rövidebb az adatküldésinél, ez esetben az adatok az elküldésig sorban állnak.

A feltétel alapú frissítés esetén a nem változó értékek elküldésének mellőzése jelentős sávzsélesség-megtakarítással jár, ami különösen igaz, ha a változások lényegesen ritkábbak, mint a lekérdezési frekvencia (például egy ajtó állapotának figyelésénél). Ez a módszer jelentős számítási erőforrást és üzenetváltást takarít meg a szerver és kliens között.

Még akkor is lehet, hogy a feltételes frissítés a legjobb megoldás, ha a jel sűrűbben változik, mint a lekérdezési intervallum, és kritikus jelről beszélünk. A hálózati torlódás azonban gondot okozhat, ha hirtelen nagy mennyiségű adatot akarunk átvinni a hálózaton. Analóg jelek esetén a torlódásokat valamelyest enyhíteni lehet holt-sáv beállításával, vagy egy megfelelő előfeldolgozási algoritmussal. Gyorsan változó, de nem kritikus adatok – például egy folyadék hőmérséklete – lekérdezéséhez továbbra is megfelelő lehet a klasszikus eljárás.

A legtöbb OPC UA-szerver lekérdezés alapú protokollt alkalmaz, például a Modbus, hogy adatokhoz jusson az I/O-eszközökről. Ha mindkét fajta lekérdezési forma elérhető, akkor a tag-eket négy csoportba osztva eljuthatunk minden jel legmegfelelőbb megjelenítési módjáig (3. ábra).

Könnyen érthető tag-nevek

A legtöbb OPC-szervernél a tag nevében szerepelnie kell a kommunikáció típusának (pl. Ethernet vagy soros), ezt követően az eszköz nevének, majd az I/O-pont nevének. Például egy szivattyú állapota így érhető el: Ethernet.Device.Pump_Status. Ebben azonban nincsen

jelölve a szivattyú helye. Mivel a SCADA akár több ezer tag-et is kezelhet, ezért az operátorok sokszor a név alapján, egy részletes leírást tartalmazó Excel munkafüzetben keresik ki a berendezés helyét.

A probléma kezelésének egyik módja, hogy szerepeltetjük a tag névben a helyet is. Két ugyanolyan szivattyút pedig az alábbi módon célszerű elnevezni: PumpA, PumpB. A könnyű megkülönböztetés érdekében tehát: Ethernet.Device_SiteA.Pump_Status és Ethernet.Device_SiteB.Pump_Status.

De miért szükséges a tag nevet a kommunikációs csatornával kezdeni? Ha az elnevezések az adott alkalmazás felépítését követik, logikailag könnyebben áttekinthető rendszert kapunk. Az 4. ábrán szemlélítjük az elnevezések felépítésének kétféle megközelítését. A bal oldali ábra zavaros, hiszen azt az érzetet kelti, hogy az A helyszínen csak ethernetes eszközök vannak alkalmazva. A jobb oldali ábrán ugyanez a rendszer látható, alkalmazás szerinti struktúrában. Ebben az esetben a helyszínmegjelöléssel kezdődik a címke elnevezése, úgy, mint SiteA.Device.Pump_Status és SiteB.Device.Pump_Status. Ezzel a lépéssel könnyebben értelmezhető címkéket kaptunk, ami a SCADA-ban való beállítást is leegyszerűsíti.

Az OPC UA egyszerűbbé teszi a távoli kapcsolódást

Az OPC-kapcsolatok konfigurálása távoli OPC-szerverrel az OPC UA-szabvány megjelenése előtt igencsak bonyolult volt. Például egy felhasználónak ugyanazzal a „felhasználónév-jelszó” párossal kellett belépnie a szerver és kliens gépeken. Ezen túlmenően a felhasználónak konfigurálnia kellett a DCOM biztonsági elemeket. Az OPC UA ezzel szemben TCP-alapú UA bináris protokollt alkalmaz adatcserére, amely a tűzfalon egyetlen port megnyitásával engedélyezhető. A felhasználók létrehozhatnak számos, egyedi porttal rendelkező TCP URL-t az OPC-szerver végpontjaihoz. A kliensek számára a szerverhez való csatlakozáshoz elegendő az URL.

Az integrált biztonsági mechanizmusok, mint például az X509-tanúsítványok, a biztonságos kommunikáció lehetőségét nyújtják az Interneten keresztül.

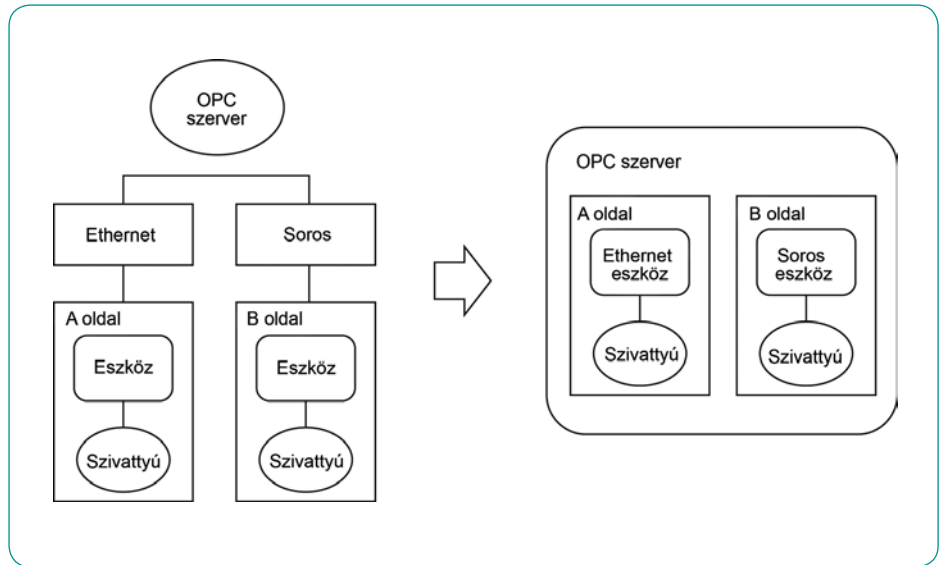
Ehhez mindössze be kell importálnunk a szerver tanúsítványait a kommunikáció hitelesítéséhez.

A szerverkereső funkcióval az OPC UA-kliensek feltérképezhetik az adott hálózaton lévő elérhető szervereket.

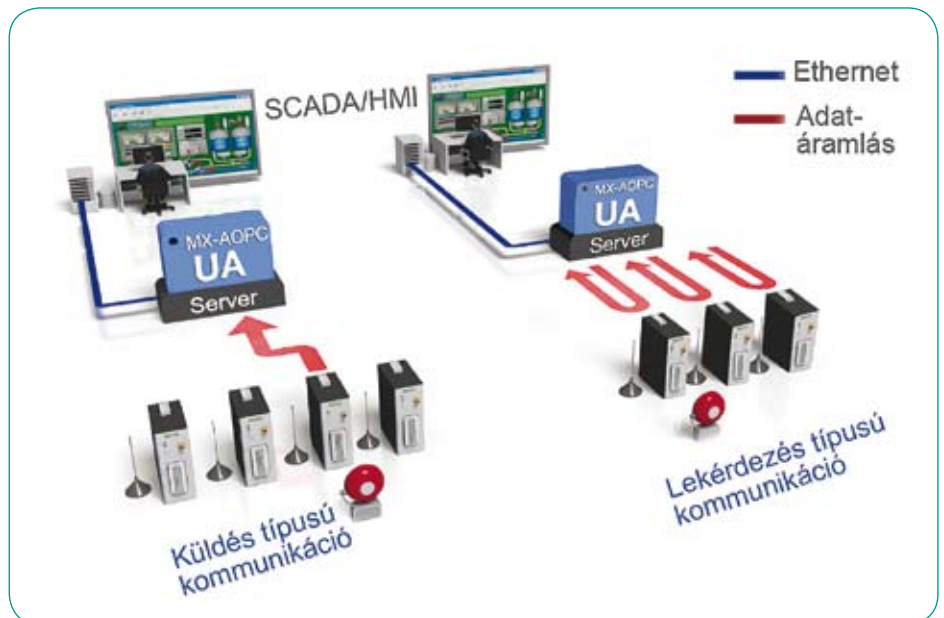
Végül pedig TCP URL kiválasztásával határozhatják meg a felhasználók, hogy melyik szerverhez kívánnak csatlakozni.

A Moxa MX-AOPC UA-szerver megoldása

Az MX-AOPC UA-szerver a Moxa szabadalmaztatott „Active OPC” monitoringtechnológiáján alapul, magában foglalja a Modbus-protokollt, ezzel nyújtva biztonságos és megbízható átjárást az eszközök és a SCADA között. A Moxa push-alapú I/O-feldolgozása



4.ábra A hagyományos és az alkalmazás logikáját követő tag-név elnevezések



5.ábra A „Push” és „Pull” típusú kommunikáció

úttörőnek számít az automatizálásban. A szabadalmaztatott MX-AOPC UA mindkét kommunikációs metódust támogatja (5. ábra).

Az MX-AOPC UA-szerver felhasználó- és alkalmazásorientált. Létrehozhatunk eszközcsoportokat, például „SiteA”-t és „SiteB”-t, ezzel az ugyanolyan nevű eszközöket külön csoportba tudjuk sorolni, miáltal a kialakult tag-nevek sokkal könnyebben értelmezhetőek lesznek.

A Moxa az adatgyűjtő termékek széles választékát kínálja

A Moxa kivételesen széles választékkal rendelkezik az ipari adatgyűjtők és a könnyen kezelhető, ipari felhasználásra alkalmas szoftverek terén.

Com-Forth Kft. - MOXA képviselő

1134 Budapest, Róbert Károly krt. 82-84.

Tel.: +36 1 413 7199, fax: +36 1 321 3899

E-mail: moxa@moxa.hu

www.moxa.hu

