OpcDbGateway – Ahoj svet

OpcDbGateway[1] je softvérová aplikačná platforma pozostávajúca z konfigurátora a výkonnej aplikácie. Konfigurátor slúži na tvorbu, ladenie a testovanie vytvorených aplikácii.

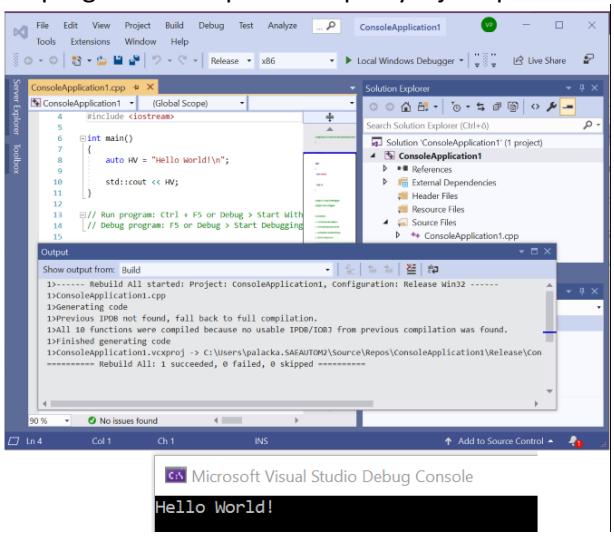
Podobne ako v iných vývojových prostrediach môžeme si prvé kroky vyskúšať na aplikácii "Hello World". Ukážeme si analógiu s tvorbou a ladením aplikácie vytvorenej v C++:

```
#include <iostream>

int main()
{
	auto HW = "Hello World!\n";
	std::cout << HW;
}
```

Vytvoríme si reťazcovú premennú HV obsahujúcu nápis "Hello World!\n".
Pomocou príkazu std::cout << HV pošleme obsah tejto premennej na štandardný výstup.
Program preložíme a spustíme konzolovú aplikáciu.

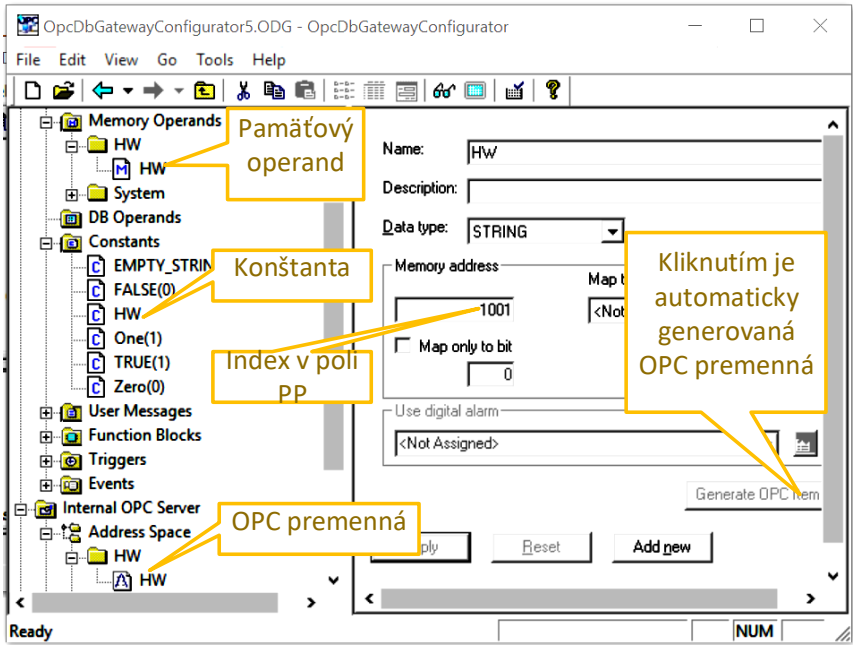
Pre tvorbu a ladenie takéhoto programu nám poslúži napr. vývojové prostredie Visual Studio 2019.



Obrázok 1 Vytvorenie, preklad a spustenie aplikácie "Hello World" v prostredí Visual Studio 2019

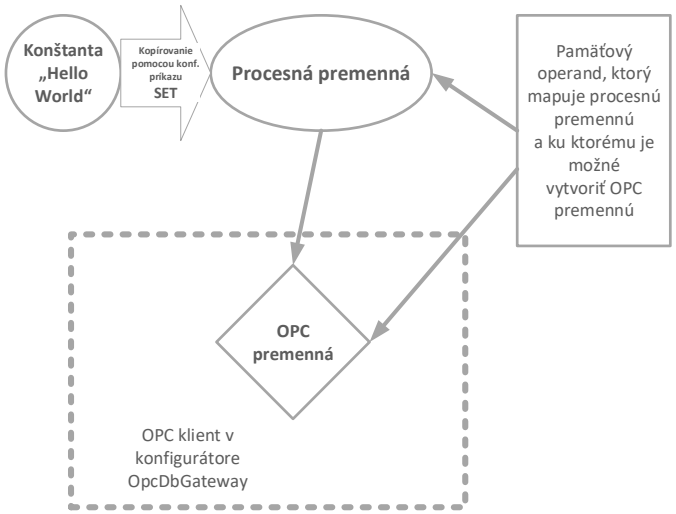
[1] Funkcionalita popísaná v tomto článku je rovnaká aj v produkte SAEAUT UNIVERSAL OPC Server s výnimkou databázovej funkcionality a funkcionality interného OPC klienta.

Teraz si ukážeme vytvorenie a testovanie podobnej aplikácie v konfigurátore OpcDbGateway. Výsledkom spustenia predchádzajúcej aplikácie bolo zobrazenie konštantného reťazca v konzolovej aplikácii. Výkonná aplikácia OpcDbGateway však nie je konzolová aplikácia ale **serverová aplikácia bez užívateľského rozhrania**, ktorá poskytuje dáta klientskym aplikáciám alebo ich ukladá do databáz a súborov. Vo Visual Studiu sme si nechali vypísať reťazec v „Debug Console“. Ako náhradu takejto konzoly môžeme pri ladení programu použiť konfiguračnú aplikáciu OpcDbGateway, ktorá funguje aj ako **klientska aplikácia komunikujúca cez OPC DA rozhranie s OPC DA serverom výkonnej aplikácie.** OPC DA server výkonnej aplikácie poskytuje prístup k tzv. **procesným premenným (PP),** ktoré môžu obsahovať premenné rôznych typov včítane reťazcov. V typických aplikáciách OpcDbGateway sa do PP ukladajú dáta z externých zariadení a aplikácii komunikujúcich s výkonnou aplikáciou OpcDbGateway. PP sú v pamäti výkonnej aplikácie uložene v poli štruktúr (tie obsahujú okrem hodnoty aj typ, časovú značku a kvalitu) a sprístupnené prostredníctvom indexov tohto poľa (Obrázok 2). Pri vytváraní konfigurácii potrebujeme zabezpečiť prehľad o použití jednotlivých PP v jednotlivých častiach aplikácie. Pre tento účel slúžia v konfigurátore tzv. **pomenované pamäťové operandy (PO)** ktoré môžu byť vytvárané v rámci stromových adresárových štruktúr a namapované na jednotlivé PP v pamäti výkonnej aplikácie. Pomocou nich vytvárame vlastne **dátový model aplikácie**. Ku každému PO je možné **jediným kliknutím vytvoriť tiež OPC premennú v adresárovej štruktúre adresného priestoru interného OPC servera**. Takto vytvorené OPC premenné poskytujú pohľad na jednotlivé PP pre OPC klientske aplikácie akou je aj konfigurátor alebo napr. webový klient SAEAUT SCADA.



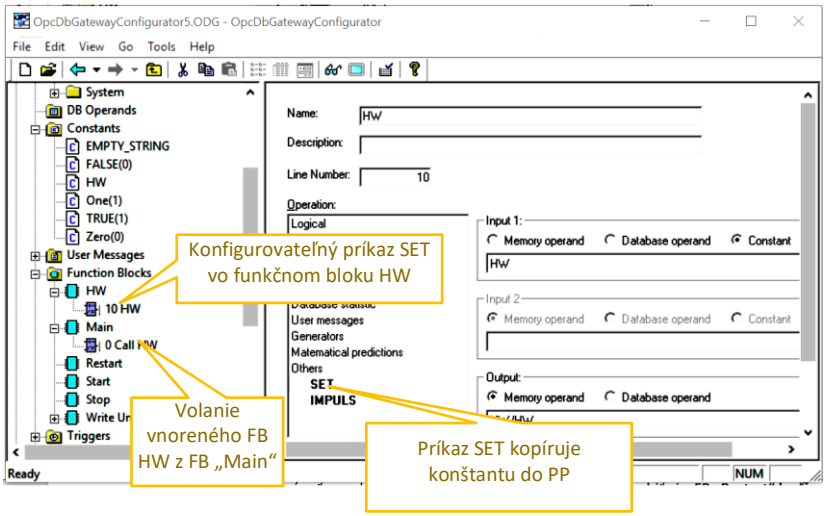
Obrázok 2 Umiestnenie pamäťových operandov, konštant a OPC premenných interneho OPC servera v konfigurácii

V OpcDbGateway implementujeme aplikáciu tak, že si nakonfigurujeme reťazcovú konštantu, ktorá bude obsahovať nápis „Hello World“. Túto konštantu skopírujeme pomocou konfigurovateľného príkazu SET do procesnej premennej. Táto PP bude namapovaná na OPC premennú, ktorej hodnotu si zobrazíme v OPC klientovi konfigurátora (Obrázok 3).



Obrázok 3 Implementácia "Hello World" v OpcDbGateway

OpcDbGateway umožňuje štrukturovať funkcionalitu aplikácie pomocou **funkčných blokov (FB),** ktoré môžu byť do seba vnorené až do 15-tich úrovní. FB (Obrázok 4) obsahujú **konfigurovateľné príkazy**, ktoré realizujú rôzne operácie nad pamäťovými operandami, databázovými operandami konštantami a užívateľskými správami. Tieto operácie môžu byť veľmi jednoduché napr. sčítanie hodnoty dvoch PP, alebo už spomenutý príkaz SET, zložitejšie napr. vykonanie nakonfigurovaného SQL príkazu nad procesnou databázou, alebo volanie **užívateľského programového modulu** (napr. PID regulátora) implementovaného v DLL**. Cyklické (synchrónne) spracovanie dát** (s konfigurovateľnou periódou) sa vykonáva v jedinom FB s názvom **„Main“**. Cyklicky sú vykonávané aj všetky FB v ňom vnorené (Obrázok 4).



Obrázok 4 Funkčný blok HW obsahuje príkaz SET, ktorým sa kopíruje konštanta HW do PP. FB HW je vnorený do FB"Main" a preto je cyklicky volaný príkazom CALL

Ak umiestnime konfigurovateľný prikaz SET do FB „Main“ bude sa tiež vykonávať opakovane. Keďže v našom prípade potrebujeme len jednorazovo skopírovať konštantný reťazec, je takáto funkcionalita redundantná. V OpcDbGateway existuje však aj niekoľko možností pre jednorazové volanie FB. Pri štarte výkonnej aplikácie sa jednorazovo vykoná FB **„Start“** alebo **„Restart“**. Ak teda umiestnime príkaz SET do FB „Start“ vykoná sa iba jeden krát. FB „Start“ sa vykoná na začiatku iba ak bola vytvorená nová konfigurácia alebo ak došlo k jej zmene. FB „Restart“ sa vykonáva pri opakovanom štarte aplikácie. Vzniká preto otázka – je potrebné umiestniť príkaz SET aj do FB „Restart“? OpcDbGateway zabezpečuje **jednoduchú perzistenciu dát** v tom zmysle, že pri ukončení aplikácie sa **stav všetkých PP uloží do databázy, z ktorej sa pri opakovanom štarte automaticky obnoví**. Keďže sme pomocou príkazu SET skopírovali konštantný reťazec do PP už v FB „Start“, nepotrebujeme to urobiť aj v FB „Restart“ pretože pri reštarte sa stav PP automaticky obnovil. Pre úplnosť treba spomenúť, že existuje ešte jeden štandardne vykonávaný FB s názvom **„Stop“,** ktorý je automaticky volaný pri ukončovaní aplikácie. Čo sa stane ak by sme príkaz SET umiestnili do FB „Stop“? Pri prvom štarte aplikácie uvidíte v OPC klientovi prázdny reťazec. Pri opakovanom štarte takejto aplikácie sa už objaví nápis „Hello World“, pretože pri predchádzajúcom štarte už bola PP nastavená v FB „Stop“.

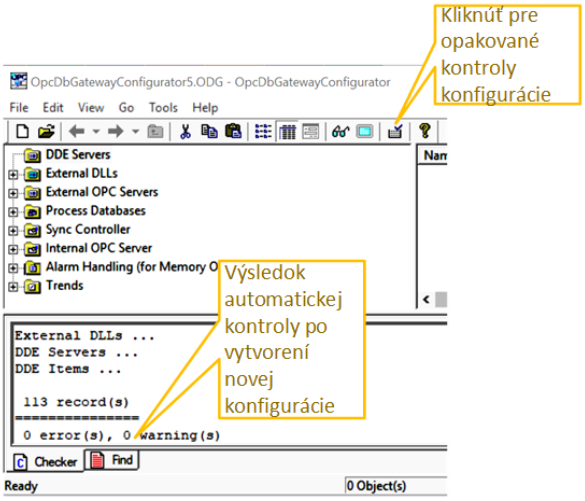
S jednorazovo vykonávanými FB sme však ešte neskončili. V skutočnosti ľubovoľný FB okrem FB „Main“ môže byť vykonaný jednorazovo. FB môžu byť totiž iniciované ako „eventy“. **Event je udalosť štartovaná nejakým „trigrom“**. Existuje viacero typov eventov (okrem volania FB) v OpcDbGateway – napr. štart externej aplikácie. Teraz sa však budeme zaoberať len eventom typu štart FB. Aktivácia eventu sa môže vykonať „trigrom“ **typu čas** – tzn. v definovanom čase, **typu hodnota** – t.j. keď vybraná PP nadobudne jednu zo zvolených hodnôt „true“ alebo „false“. Existuje tiež možnosť štartu eventu keď sú splnené obidve podmienky – **definovaný čas a súčasne definovaná hodnota PP**. Eventy majú tiež svoju definovanú **prioritu**. Ak by sa v rovnakom čase mali vykonávať dva rôzne eventy, bude poradie ich vykonávania dané ich prioritou.

I keď periodické vykonávanie FB je možné zabezpečiť v FB „Main“, pomocou periodicky štartovaného eventu je tiež možné zabezpečiť periodické volanie FB. Aký je teda rozdiel medzi periodickou funkcionalitou v „Main“ a tou, ktorá je implementovaná prostredníctvom eventov? Cyklus vykonávaný prostredníctvom „Main“ sa začína **implicitným načítaním vstupov z externých aplikácii a zariadení**, t.j. takým, ktoré nie je sprostredkované konfigurovateľnými príkazmi, pokračuje ich **spracovaním pomocou konfigurovateľných príkazov v FB „Main**“, a je zakončené **implicitným kopírovaním z PP do výstupov pre externé aplikácie.** Takýto cyklus je typicky implementovaný v priemyselných riadiacich systémoch (**PLC**) ktoré simulujú kvázi paralelné vyhodnocovanie logických výrazov. V súvislosti s OpcDbGateway ho označujeme ako **synchrónne**. Funkcionalita v FB „Main“ môže byť kombinovaná s tzv. synchrónnymi eventami, ktoré sú charakteristické tým, že „trigrovacie“ podmienky pre eventy sú vyhodnocované **vždy na začiatku periódy synchrónneho spracovania**. Naproti tomu „trigrovacie“ podmienky pre tzv. **asynchrónne eventy** sa vyhodnocujú a im zodpovedajúce eventy sa vykonávajú (v poradí podľa ich priority) takmer okamžite ako sa objavia. **Asynchrónne eventy** zabezpečujú taktiež buď vykonávanie FB, volanie externého procesu / aplikácie alebo niektoré špeciálne operácie ako napr. vytvorenie nového log súboru. Synchrónna a asynchrónna funkcionalita v OpcDbGateway sú rozdelené do **dvoch programových vlákien**. To zaručuje, že vykonávanie asynchrónnych eventov, hoci aj časovo náročnejších nenaruší dodržiavanie periódy synchrónneho spracovania.

V súvislosti s periodickými časovými trigrami je ešte užitočné spomenúť, že **je možné zadefinovať počet opakovaní**. **Čas vykonávania** môže byť definovaný ako **absolútny** alebo ako **relatívny** vo vzťahu k štartu aplikácie.

Zrekapitulujme si ako vytvoríme a budeme monitorovať vykonávanie aplikácie „Hello World“ v OpcDbGateway konfigurátore:

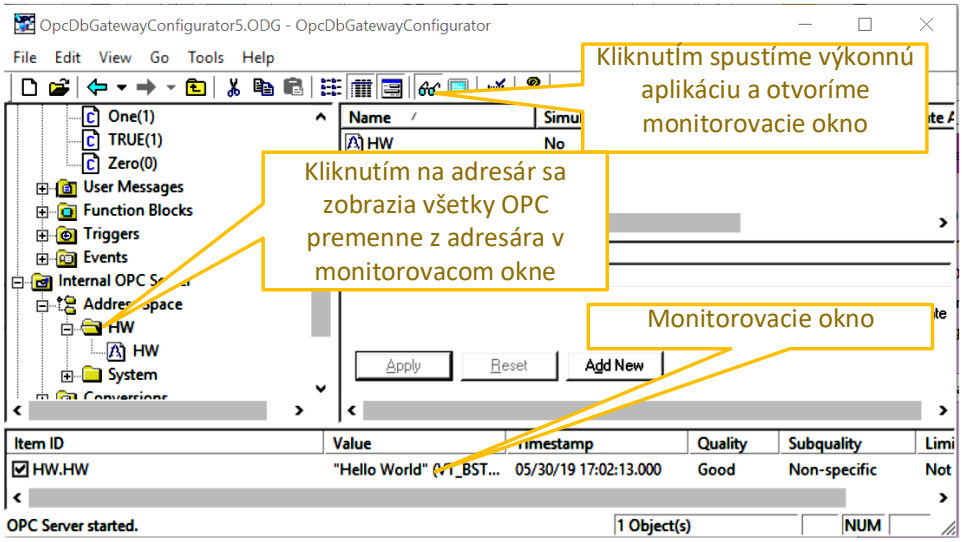
1. Vytvoríme si konfiguráciu pre novú aplikáciu príkazom v menu konfigurátora File->New. Pri tom prebehne prvá automatická kontrola konfigurácie. (Obrázok 5)
2. zadefinujeme si reťazcovú konštantu s názvom HW obsahujúcu „Hello World“ (Obrázok 2)
3. vytvoríme si PO v novom adresári s názvom HW. PO si namapujeme na OPC premennú s rovnakým menom a v rovnako pomenovanom adresári adresného priestoru interného OPC servera (Obrázok 2)
4. Zadefinujeme si FB s názvom HW, ktorý bude obsahovať jediný konfigurovateľný príkaz „SET“ pomocou ktorého je obsah konštanty HW prekopírovaný do PP ktorú referencuje PO s názvom HW umiestnený v adresári HW (Obrázok 4).
5. Spustíme kontrolu vytvorenej konfigurácie (Obrázok 5).
6. Spustíme OPC klienta v konfigurátore (ikona s okuliarmi) a v monitorovacom okne sa objaví nápis „Hello World“(Obrázok 6).



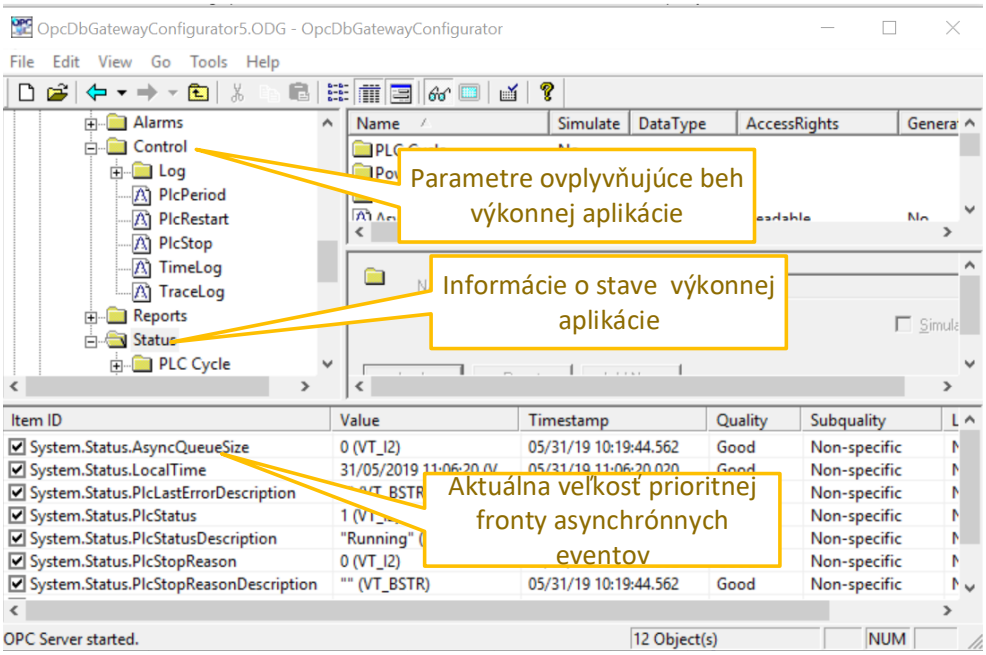
Obrázok 5 Kontrola konfigurácie po otvorení a po úpravách

V OpcDbGateway môžeme monitorovať a ovplyvňovať funkcionalitu nakonfigurovanej aplikácie nielen prostredníctvom PP a OPC premenných ktoré sú špecifické pre danú aplikáciu (Obrázok 6) ale tiež prostredníctvom **systémových PP**, ktoré sú súčasťou **automaticky vytváranej šablóny konfigurácie**. Časť týchto premenných (v adresári „Status“ - Obrázok 7) je určená len pre čítanie a časť umožňuje tiež zápis. Napr. premenná System.Status.AsyncQueSize informuje o tom koľko aktivovaných asynchrónnych eventov ešte nebolo vykonaných. Prostredníctvom premenných v adresári Control/Log s názvom TimeLog a TraceLog je napr. možné za behu meniť množstvo informácii o behu aplikácie zapisovaných do

log-súboru (Obrázok 8, Obrázok 9). Zmenu hodnoty nejakej riadiacej premennej je možné vykonať po kliknutí pravým tlačítkom myši na premennú v monitorovacom okne.



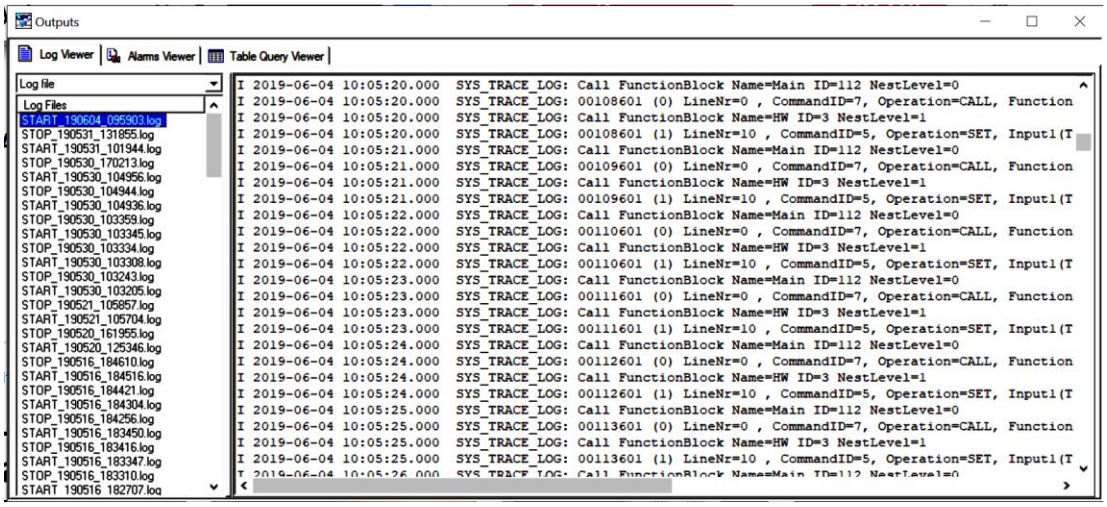
Obrázok 6 Monitoring obsahu PP prostredníctvom OPC premennej, do ktorej je mapovaná



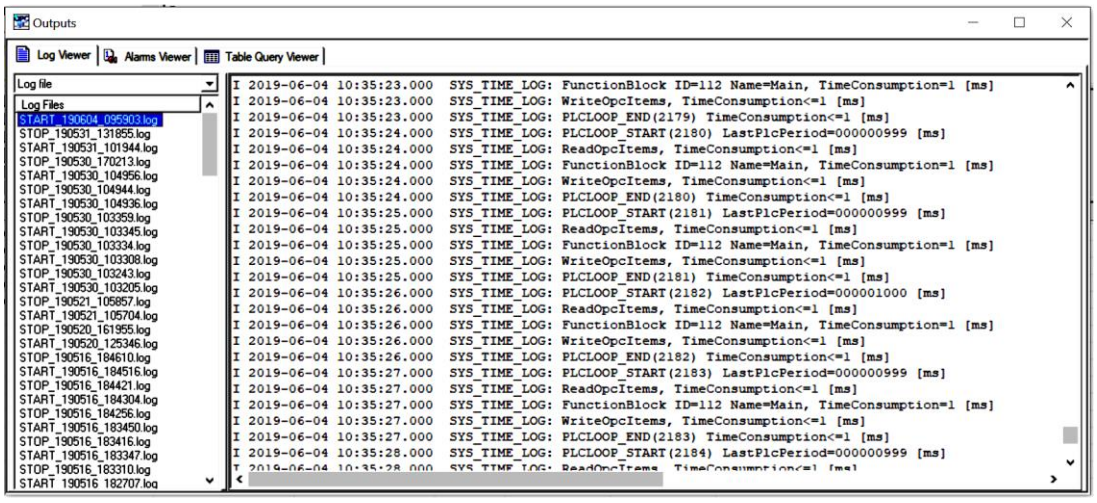
Obrázok 7 Monitoring a riadenie výkonnej aplikácie 0pcDbGateway. Premenne v adresári "Control" slúžia pre riadenie a v adresári "Status" pre monitoring.

Zobrazenie log súboru je priebežne aktualizované. Môžeme ho sledovať v konfigurátore keď vyberieme v hlavnom menu View->Output view->Log Viewer.

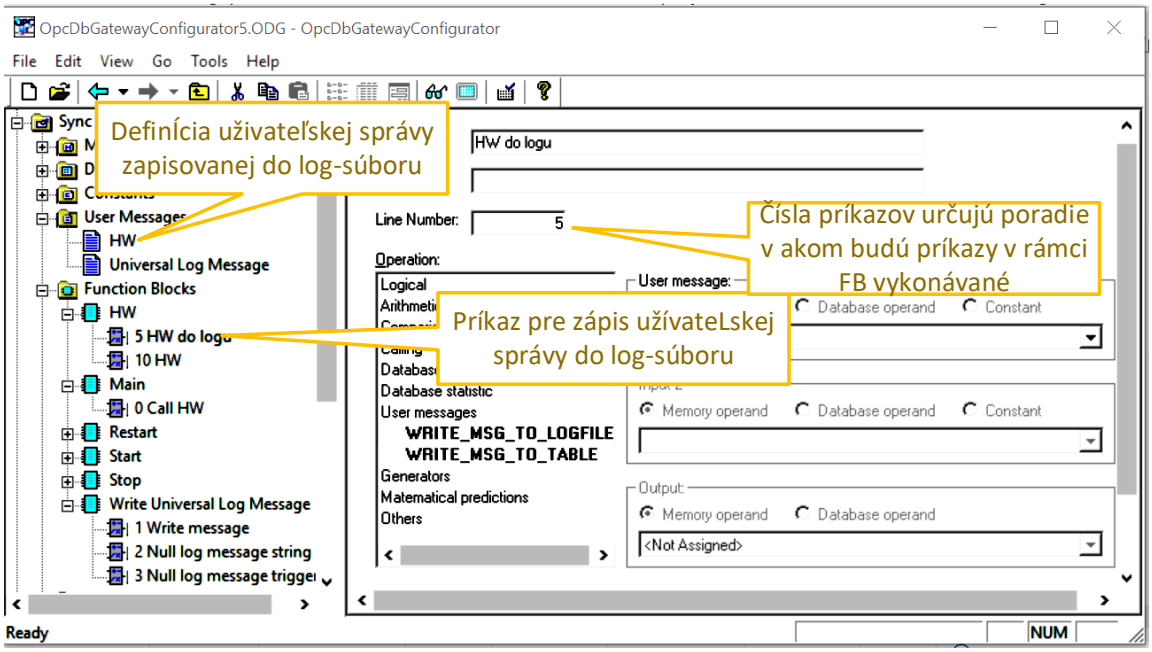
Do log-súboru môžeme zapisovať aj užívateľské správy prostredníctvom konfigurovateľného príkazu WRITE_MSG_TO_LOGFILE. Takýmto spôsobom by sme mohli rozšíriť aj našu aplikáciu. Stačí zadefinovať užívateľskú správu a nechať ju zapísať do log-súboru pomocou konfigurovateľného príkazu WRITE_MSG_TO_LOGFILE (Obrázok 10). Užívateľské správy nemusia obsahovať len text ale tiež parametre – MO, ktoré zabezpečia zápis aktuálnych hodnôt zodpovedajúcich PP do log-súboru.



Obrázok 8 Log súbor pri nastavení systémovej premennej TraceLog na true. Zaznamenávané sú údaje o vykonávaní jednotlivých funkčných blokoch, konfigurovateľných príkazoch včítane ich argumentov



Obrázok 9 Log súbor so zaznamenávaním časových údajov pre jednotlivé aktivity (systémová premenná TimeLog = true)



Obrázok 10 Rozšírenie konfigurácie o zápis užívateľskej správy do log-súboru. Poradie vykonávanie konfigurovateľných príkazov v rámci FB je dané ich číslovaním.

V rámci FB HW na obrázku Obrázok 10 máme teraz dva konfigurovateľné príkazy – „HW do logu“ s číslom 5 a „HW“ s číslom 10. Tieto čísla zabezpečujú aby vykonávanie konfigurovateľných príkazov bolo vykoné vo vzostupnom poradí. Číslovanie s medzerami sa používalo už u starého dobrého jazyka BASIC, aby bolo možne jednoduché vkladanie nových príkazov bez prečíslovania.

Z obsiahleho popisu takého jednoduchého príkladu sa môže zdať, že konfigurovanie aplikácii v OpcDbGateway je zložité. Zložitosť však spočíva skôr v tom, že sme sa pokúsili opísať aspoň časť základných funkcionalít, ktoré OpcDbGateway ponúka. Pri využití pomocných funkcionalít konfigurátora ako napr. mapovanie vstupov a výstupov z rôznych externých zariadení a databáz je však možné aj aplikácie s obrovským množstvom vstupov a výstupov konfigurovať relatívne veľmi rýchlo. Okrem toho, OpcDbGateway v sebe zahŕňa množstvo funkcionalít, pre ktoré by inak bolo potrebné nasadiť samostatné aplikácie.