TC-Konektor

TC-Konektor slúži na pripojenie a posielanie výrobných dát do Automation plus. Program PLC je napísaný pre TwinCAT2 od verzie „build 1544“. Tento PLC program vytvára a ukladá súbor s dátami pre skript, ktorý následne posiela údaje na databázový server. Spojenie z databázovým serverom zabezpečuje skript vytvorený v programovacom prostredí Autoit . Jeho úlohou je posielať dáta zo zdrojového súboru „OEE.txt“ na Databázový server. Všetky potrebne nastavenia pre spojenie zo serverom sa nachádzajú v konfiguračnom súbore. Ďalšia neoddeliteľná súčasť celého systému sa nachádza už priamo na servery. To sú už samotne databázy a tabuľky pre nastavenie TC-konektora .

1. TC-Konektor

TC-Konektor pozostáva s dvoch hlavných častí. Prvá časť TC-PLC je Implementovaná do samotného PLC Programu v TwinCAT Systéme. Tento vytvára „logovací“ súbor, ktorý následné spracúva druha časť TC-Konektora „skript“. Skript zabezpečuje kontrolu a roztriedenie údajov pre databázu (poprípade databázy podľa konfigurácie). Následne po roztriedení posiela údaje v správnom tvare na Databázový server. A po odoslaní zabezpečuje vymazanie odoslaných údajov zo súboru. Implementácia TC-Konektora na jednotlivý stroj pozostáva v niekoľkých krokoch

* Spojazdnenie Automatického nastavenia dátumu a času podľa časového servera
* Spojazdnenie TC-PLC časti
* Napojenie TC-PLC časti na PLC Stroja
* Úprava konfiguračného súboru pre skript
* Zabezpečenie automatického spúšťania Skriptu
  1. Potrebne súbory

TC-Konektor pre správnu činnosť potrebuje mat plný prístup k nasledovným priečinkom a adresárom:

* „C:\Twincat\OEE.txt”
* „C:\Data\TC\_Konektor\Data\konfig.ini”
* „C:\Data\TC\_Konektor\TC\_Konektor.exe”
* „C:\Data\TC\_Konektor\libmysql.dll”
* „C:\Data\TC\_Konektor\Data”

Ďalej potrebujeme TwinCAT PLC projekt stroja a TC-PLC projekt aby bolo možné spojiť jednotlivé PLC projekty do funkčného celku. A v neposlednom rade samotný AutoIT skript s konfiguračným „ini” súborom.

1. TwinCAT TC-PLC

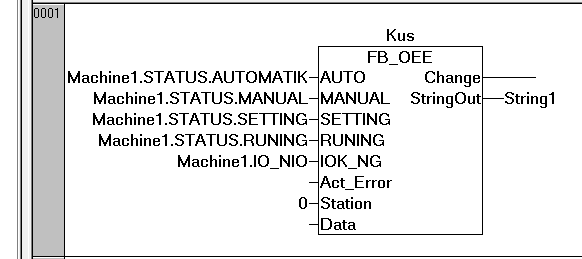
Program PLC ma zisťuje základne stavy PLC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Signál** | | **Dĺžka** |  |  |
| Status stroja | | 1byte |  |  |
|  | Automatika | 1bit |  |  |
|  | Nastavenie | 1bit |  |  |
|  | Ručne | 1bit |  |  |
|  | Automatika Beží | 1bit |  |  |
| IO\_Nio | | 16bit |  |  |
|  | OK | 1 bit |  |  |
|  | NG1 | 1 bit |  |  |
|  | NG2 | 1 bit |  |  |
|  | NG3 |  |  |  |
|  | NG4 |  |  |  |
|  | NG5 |  |  |  |
|  | NG6 |  |  |  |
|  | NG7 |  |  |  |
| Alarm | | 32 Byte |  |  |
| HelpBits | | 16 bit |  |  |

PLC program si zo systému vytiahne  časovú značku „Dátum a čas“ a priradí k nemu zmenu stavu vstupných parametrov. Následné tento reťazec uloží do súboru „C:\Twincat\OEE.txt“.

* 1. PLC Program

TC-PLC sa skladá z dvoch funkčných blokov a hlavného „Main programu“. Main program je upravitelný (priradenie vstupných hodnôt pre Automatiku pop. Chybu stroja) a musia v ňom byt volané oba funkčné bloky „FB\_OEE“ a „FB\_ADD\_DATA“ a to v poradí ako sú napísané (FB\_OEE vytvára údaje pre FB\_ADD\_DATA).

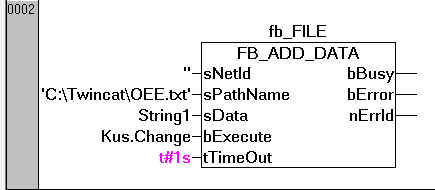
* + 1.  Funkcia FB\_OEE

Funkcia ma za úlohu priradiť k aktuálnej časovej značke, získanej zo systému Windows. ,pripojiť údaje zo vstupných hodnôt funkcie. Spojene údaje sú uložené do reťazca oddelene znakom “|“ postupnosť údajov je nasledovná

CasovaZnacka|CisloStanice|StatusStroja|OK\_NOK|AktualnaChyba|PredchadzajucaChyba| <EOL>

Napr.: '2017-06-14-23:12:21.141|0|8|-1|0|0| $N'

Pri novom stave vstupov (zmena vstupnej hodnoty )sa na dĺžku jedného cyklu zmení stav výstupu funkcie FB\_OEE.Change na log 1 pre funkčnosť zápisu nasledovnej funkcie(aby nedochádzalo k nepretržitému zápisu do súboru

* + 1. Funkcia FB\_ADD\_DATA

Úlohou tejto funkcie je zapísať vstupne údaje do súboru sPathName.

* + 1. MAIN
  1. Konfigurácia
  2. Funkcie

1. Skript TC-Konektor

Skript TC-Konektor je napísaný v Skriptovacom prostredí AutoIT. TC-Konektor je program na posielanie údajov uložených do textového súboru, na server Mysql. Údaje uložené v súbore musia byt v presne definovanom tvare a postupnosti . Súbor s dátami sa najskôr premiestni na iné miesto a potom ho skript zanalyzuje a pošle údaje na server .

* 1. Funkcia skriptu

Skript pozostáva z viacerých časti. Postupnosť je z inicializačného súboru sa získajú nastavenia servera a zapíše sa do neho „Hardwarové číslo“( $SerNo) stroja (MAC Adresa prvej karty). Potom sa pokúsi pripojiť na server s inicializačnými dátami na databáze konfig. Po úspešnom pripojení zisťuje prítomnosť záznamu so „$SerNo“ v Tabuľke Phoenix . A po úspešnom nájdení záznamu získa nove prihlasovacie údaje pre daný stroj. V danej tabuľke Phoenix sa zaktualizujú niektoré údaje (Aktuálna IP, časová značka, Typ konektora, Firmvare, verzia konektora, sériové číslo, Mac Adresa). Po tejto aktualizácii záznamov si skript vypýta dátum a čas zo servera a zároveň prestaví Dátum a čas na lokálnom stroji. Nasleduje ukončenie komunikácie pod inicializačnými údajmi. Teraz sa skript pokúsi pripojiť k serveru z údajmi získanými z Databáze konfig a tabuľky Phoenix. Po pripojení je skryt pripravený prenášať dáta na server . Preto sa musí ale najskôr presvedčiť čí ma čo prenášať teda potrebuje získať vstupne údaje zo súboru, Preto najskôr zisťuje prítomnosť súboru dát na dočasnom ukladacom mieste (@scriptdir/Data/OEE.txt).Ak sa tam nachádza súbor a nie je prázdny. Skript z neho najskôr vyčíta údaje a následne pošle na server. Keď sa v dočasnom priečinku súbor nenachádza premiestni si tento súbor zo zdrojového priečinka do dočasného . A následne ho znova overí (Veľkosť..).Nasleduje odoslanie údajov na server. Až po úspešnom odoslaní celého súboru sa súbor vymaže a proces z kopírovaním a overovaním sa opakuje.

Pri chybe v spojení zo serverom alebo načítaní údajov ... sa automaticky generuje logovací súbor (@scriptdir/Data/ Error.log)

* 1. Konfigurácia

Konfigurácia TC-konektora pozostáva dvoch častí. Prvá časť je na strane skriptu a druha cast na strane servera. Ani jedna časť nie je úplná bez tej druhej. Prvá časť sa stará o pripojenie na server a pripojenie ku konfiguračnej databáze (konfig). Ďalej ju tam záznam o hardwery na ktorom skript bežal, bezpečnostná poistka proti neoprávnenému prepisu zo stroja na iný stroj. Ďalej sú tam zdrojové cesty k súborom z dátami na odoslanie. A v neposlednom rade konfigurácia pre viacstrojový skript. Pre lepšie pochopenie jeden Industrial PC na dva stroje - dve strany stroja môžu vyrábať rozdielny diel a...