

Izdelava operaterskega vmesnika sistema za krmiljenje talnih transporterjev

Urban Perme

Mentor: izr. prof. dr. Gašper Mušič

Univerza v Ljubljani, Fakulteta za elektrotehniko

Tržaška 25, 1000 Ljubljana

urbanperme@yahoo.com

Implementation of an operator interface for a floor conveyor control system

The article is about development of a HMI (human machine interface) supervisory system for control of four floor conveyors. The basic procedures for developing HMI device programs using WinCC flexible 2008 software are described. Such devices are used in all areas of industrial automation. The basic procedures of programming using WinCC flexible, which is Siemens's software for programming Siemens's HMI units, are described. This software tool covering a wide range of different functions, which are necessary for the production of high quality and as simple as possible human-machine interface. This range of functions include in particular: productions of screens, creating alarms, linking variables, PLC and HMI unit communication and all other necessary functions. The main purpose of this project was therefore to create the most easy and flexible control system, which will include all the requirements of control transporters, show events in the system and effectively displaying and debugging errors.

Kratek pregled prispevka

Članek govori o izdelavi nadzornega sistema HMI (vmesnik človek-stroj) za krmiljenje štirih talnih transporterjev. Opisani so temeljni postopki izdelave programa za naprave HMI, ki jih uporabljamo na vseh področjih avtomatizacije v industriji, s pomočjo programskega orodja WinCC flexible 2008. Opisani so osnovni principi programiranja v WinCC flexible, ki je torej programsko orodje za programiranje Siemensovih enot HMI. To programsko orodje pokriva širok spekter najrazličnejših funkcij, ki so potrebne za izdelavo kvalitetnega in hkrati kolikor mogoče enostavnega vmesnika človek-stroj. V ta spekter funkcij spadajo predvsem: izdelava zaslonov, kreiranje alarmov, povezava spremenljivk, komunikacija PLK in enote HMI ter vse ostale potrebne funkcije. Osnovni namen tega projekta je bil torej ustvariti čim bolj enostaven in fleksibilen nadzorni sistem, v katerem bodo zajete vse zahteve krmiljenja transporterjev, prikaz dogajanja na sistemu ter hitro in učinkovito prikazovanje ter odpravljanje napak.

1 Uvod

Nadzorni sistemi HMI predstavljajo vmesnik med procesom in operaterjem [1]. To zajema nekaj osnovnih in najpomembnejših opravil:

- **Vizualizacija procesa**
Proces je simbolično prikazan na ekranu HMI-naprave in se dinamično posodablja. Posodabljanje je neposredno odvisno od stanja procesnih spremenljivk.
- **Kontrola operaterja nad procesom**
Operater izvaja kontrolo nad procesom. To pomeni, da lahko vnaša vrednosti procesnih spremenljivk, jih spreminja, prednastavlja itd.
- **Prikazovanje alarmov**
Kritična stanja vrednosti spremenljivk znotraj procesa avtomatično sprožijo alarm. Ti alarmi morajo vsebovati dovolj informacij, da lahko napako na sistemu lociramo in odpravimo.
- **Arhiviranje procesnih spremenljivk (tagov) in alarmov**
Če hočemo imeti vpogled v preteklost dogajanja, moramo imeti arhiv. WinCC flexible omogoča izdelavo arhivov, v katerih so zapisana stanja spremenljivk in sproženih alarmov.

Glavna tema tega članka je torej izdelava nadzornega sistema za krmiljenje štirih talnih transporterjev. Gre za transporterje v avtomobilski industriji, ki skozi proizvodnjo prenašajo gotova vozila, na katerih delavci opravljajo različna zaključna dela.

Transporterji so krmiljeni delno preko tipk na elektro omari (zagon, zaustavitev, potrditev napake,...), ostali del nadzora pa se izvaja preko enote HMI na operatorskem pultu. Uporabili smo 6-palčno, na dotik občutljivo HMI-napravo SIMATIC MP 177 [2], ki je prikazana na sliki 1. Ta enota je že sestavni del operatorskega pulta in je povezana z glavnim krmilnikom preko omrežja PROFINET.



Slika 1: Enota HMI SIMATIC MP 177

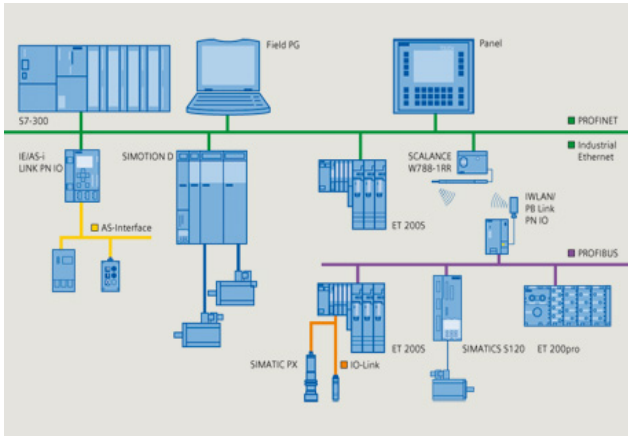
Na HMI-enoti imamo vse potrebne kontrolne tipke za krmiljenje transporterjev, možnost vpisovanja in spremljanja hitrosti vseh transporterjev in nekaj sinoptičnih prikazov. Sinoptični prikazi, ki so manjšega obsega kot pri nekaterih drugih nadzornih sistemih, nam omogočajo hiter vpogled v dogajanje na samem sistemu. Za diagnostiko in odkrivanje napak pa imamo narejen tudi prikaz vseh napak in alarmov ter vse potrebne informacije o napaki.

2 Izvedba vodenja sistema

2.1 Izvedba krmilja

Način krmiljenja procesa je lahko izveden kot centralni ali oddaljeni sistem. Za centralni sistem je značilno, da so vsi elementi krmilja, ki so vključeni v proces, speljani na eno centralno mesto, ki se običajno nahaja v glavni krmilni omari. V tej omari se nahaja krmilnik (CPE) z vsemi potrebnimi vhodno/izhodnimi moduli. To pogosto pomeni zelo veliko ožičenja, ki je cenovno dražja odločitev predvsem v sistemih, kjer so senzorji in aktuatorji zelo oddaljeni od krmilnika.

Če imamo senzorje in aktuatorje zelo oddaljene od krmilnika, lahko postane ožičenje zelo obsežno in zapleteno. Težave lahko nastanejo tudi zaradi številnih elektromagnetnih in drugih motenj. V tem primeru se poslužujemo oddaljenih sistemov.



Slika 2: Oddaljeni sistem v omrežju PROFIBUS ali PROFINET

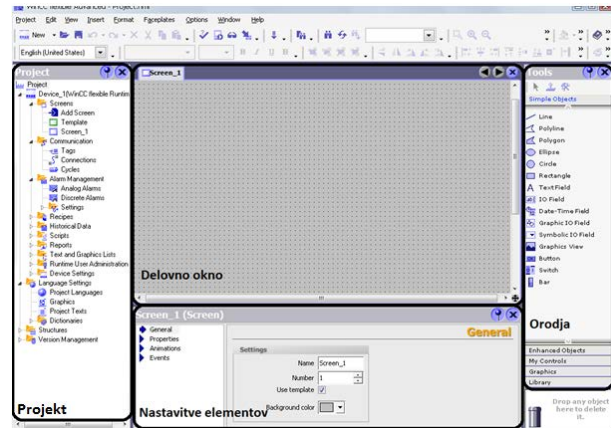
Oddaljeni sistemi (slika 2) so povezani preko omrežja PROFIBUS ali PROFINET. To pomeni, da imamo centraliziran krmilnik, ki komunicira z oddaljenimi (distribuiranimi) enotami. Te enote so največkrat blizu samega procesa in so na njih neposredno priključeni senzorji in aktuatorji. S tem imamo do centralnega krmilnika komunikacijo samo preko enega kabla.

Mi smo uporabili enote SIMATIC ET 200S, ki smo jih opremili z ustreznimi moduli za vodenje posameznega transporterja. Vsak transporter ima svojo enoto ET 200S, ki je preko omrežja PROFINET povezana z glavnim krmilnikom, ki pa se nahaja v operaterskem pultu, v katerem je tudi enota HMI. Vsa senzorika, ki je potrebna za vodenje posameznega transporterja, je povezana na enoto ET200S.

2.2 WinCC flexible 2008

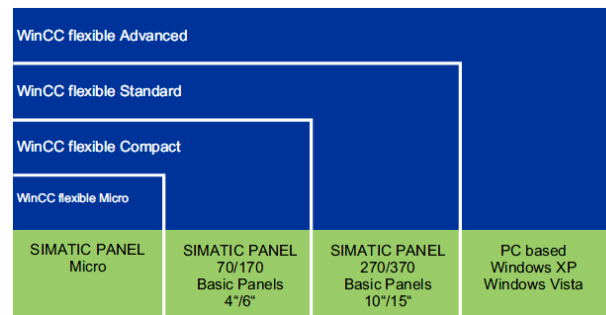
Program za enoto HMI je narejen s pomočjo programskega orodja WinCC flexible 2008 [3] (slika 3). S tem orodjem lahko izdelamo sisteme za vse naprave HMI od najmanjših HMI Micro Panels do najzmogljivejših HMI Multi Panels. Orodje je primerno tudi za izdelavo nadzornih sistemov, ki temeljijo na industrijskih računalnikih. Operacijski sistem, na katerem deluje WinCC flexible, je največkrat kar MS Windows XP, vendar mora biti različica SP3 (Service Pack 3) ali novejša. Povezovanje z aplikacijami Step 7 preko MPI, PROFIBUS,

PROFINET in drugih omrežij je mogoče preko strežnika OPC (OLE for process control), ki nam zagotavlja kompatibilnost z veliko večino strojne opreme, ki jo uporabljamo, brez dodatnih gonilnikov.



Slika 3: Osnovno uporabniško okno v WinCC flexible

WinCC Flexible [3] obstaja v več različicah (slika 4). Na voljo imamo vse od WinCC flexible Micro pa do različice WinCC flexible Advanced. Vsaka višja različica razširi spekter podprtih naprav HMI in funkcionalnosti, ki jih orodje omogoča.



Slika 4: Različice orodja WinCC flexible

2.2.1 Izdelava zaslonov

Vsaka naprava HMI ima določeno velikost zaslona. Od te velikosti je odvisna tudi velikost zaslona, ki je na voljo v programu. Seveda moramo HMI-napravo, ki jo uporabljamo, izbrati na začetku projekta.

Vsak zaslon vsebuje aktivne in pasivne elemente, ki jih dodajamo po potrebi. Oblike so lahko privzete, določene elemente pa lahko

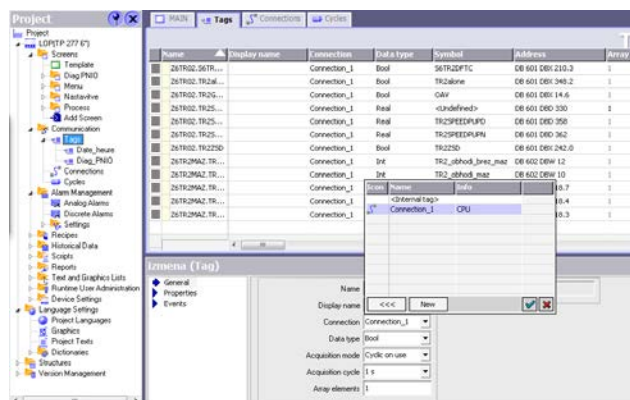
oblikujemo tudi sami. Aktivni elementi so tisti, s pomočjo katerih lahko vplivamo na proces (gumbi, vnosna polja), ali pa s svojim spreminjanjem stanja podajajo informacije o procesu (nivojski histogrami, spreminjanje barv posameznih likov). Pasivni elementi pa so ponavadi posamezni liki, ki so ves čas enaki (enaka oblika in enaka barva), ne glede na to, kaj se dogaja v procesu. Takih elementov je ponavadi na sinoptičnih prikazih bolj malo. Z njimi največkrat prikazujemo različne elemente v okolici procesa za lažjo orientacijo in v samem procesu ne igrajo bistvene vloge.

2.2.2 Spremenljivke

V WinCC flexible [3] za komunikacijo z ostalimi programi (Step7) uporabljamo spremenljivke (tage). Te spremenljivke delimo na zunanje in notranje.

Zunanje spremenljivke uporabljamo za komunikacijo HMI-naprave s krmilnikom. Te spremenljivke so shranjene na pomnilniških lokacijah krmilnika in so v bistvu preslikane iz Step 7 v WinCC flexible (slika 5). Tako imamo do njih dostop iz krmilnika in HMI-naprave. Podatkovni tipi teh spremenljivk so odvisni od krmilnika. Tiste podatkovne tipe, ki jih podpira krmilnik, podpira tudi HMI-naprava.

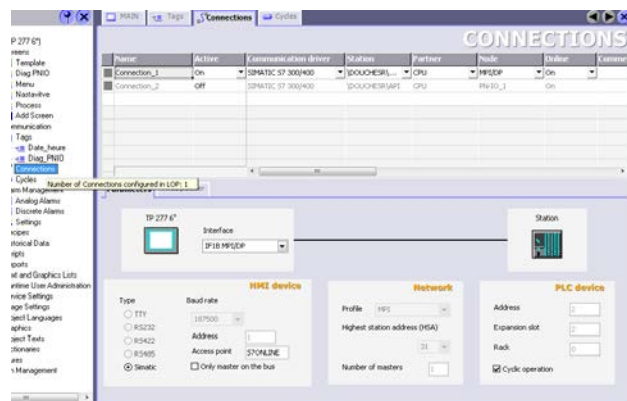
Notranje spremenljivke pa nimajo nobene povezave s PLC-jem. Shranjujejo se v pomnilniku HMI-naprave, ki ima direkten dostop do njih. Uporabljamo jih za razne lokalne izračune, ki jih potem prikazujemo na zaslonu nadzornega sistema. Za notranje spremenljivke lahko uporabljamo vse podatkovne tipe.



Slika 5: Seznam spremenljivk v WinCC flexible

2.2.3 Komunikacija s krmilnikom

Da bi si dve napravi lahko izmenjevali podatke, mora med njima potekati komunikacija. V tem primeru govorimo o komunikaciji med PLC-jem in HMI-napravo (slika 6). Če HMI-napravo predhodno določimo že kar v projektu v Step 7, nam WinCC flexible [3] povezavo uredi že kar sam. V našem primeru je bila komunikacija vzpostavljena preko omrežja PROFINET [4], ki uporablja komunikacijski protokol TCP/IP. To pomeni, da je potrebno vsaki napravi določiti IP-naslov. V tem primeru pa moramo paziti, da se naslovi ustrezno ujemajo, saj nam jih program ne bo samodejno nastavil.



Slika 6: Komunikacija PLC-ja in enote HMI

2.2.4 Alarmi

Kadar govorimo o alarmih v WinCC flexible [3], imamo v mislih procesne spremenljivke, ki jih kontroliramo iz različnih vzrokov. Če imamo na primer šaržni proces v nekem reaktorju in dodajamo različne snovi v reaktor, so ponavadi predpisane dovoljene količine posameznih snovi. Preko senzorjev, ki jih imamo v sistemu, dobivamo podatke bodisi v analogni, bodisi v digitalni obliki. Spremenljivke, ki so povezane s tem senzorjem, moramo v nadzornem sistemu definirati kot alarme, da bi ob izrednih razmerah lahko javile napako na sistemu.

Imamo torej dve vrsti alarmov:

- **Analogni alarmi** so povezani z analognimi spremenljivkami in se sprožijo, kadar je presežena določena vrednost neke veličine, ki jo merimo. Če merimo hitrost motorja, se alarm sproži, kadar hitrost preseže neko dovoljeno vrednost.
- **Diskretni alarmi** zaganavajo spremembo stanja v procesu. Tako kot analogni alarmi so povezani s spremenljivkami, vendar z binarnimi spremenljivkami (Bool). Te vrste alarmov najpogosteje uporabljamo, saj z njimi lahko nadzorujemo vse vrste odklopnikov in raznih zaščit, ki zavzemajo samo dve stanji (0 ali 1). Tako lahko naprimer definiramo alarm za kontrolo odprtosti ventila (dve stanji).

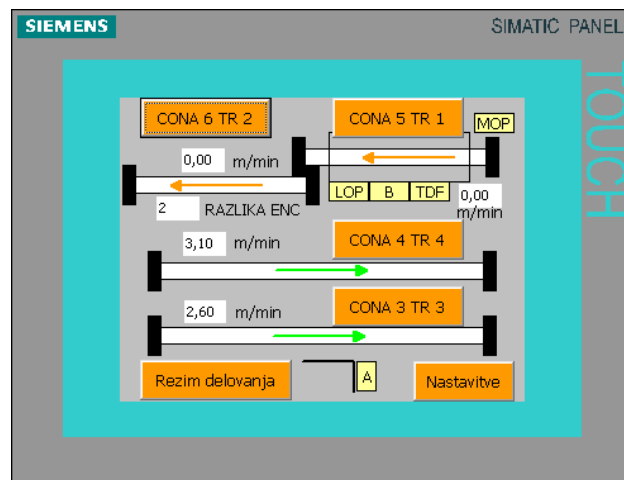
Za vsak alarm moramo definirati spremenljivko, ki ga bo sprožila. Kadar želimo kreirati veliko diskretnih alarmov, ponavadi določimo neko spremenljivko velikosti bajta (byte) ali pa besede (word) in njene bite povežemo s signali (spremenljivkami), ki jih želimo nadzorovati. Če naprimer želimo nadzorovati prisotnost desetih odklopnikov, definiramo zunanjo spremenljivko »Odklopniki« velikosti besede (16 bitov) in nato v Step7 ustrezno povežemo signale s to spremenljivko. Biti te spremenljivke bodo nato naš sprožilec (trigger) za posamezen alarm.

Vsakemu alarmu je treba določiti tudi besedilo. To besedilo se prikaže na zaslonu za alarme v trenutku, ko je alarm sprožen. Enako velja za analogne alarme.

2.3 Nadzorni sistem HMI za krmiljenje talnih transporterjev

Struktura našega nadzornega sistema je drevesna. To pomeni, da imamo osnovni sinoptični prikaz vseh transporterjev na glavnem zaslonu (slika 7), ki se prikaže ob zagonu sistema. Do ostalih zaslonov, ki jih še imamo,

pa pridemo preko glavnega zaslona na več načinov.



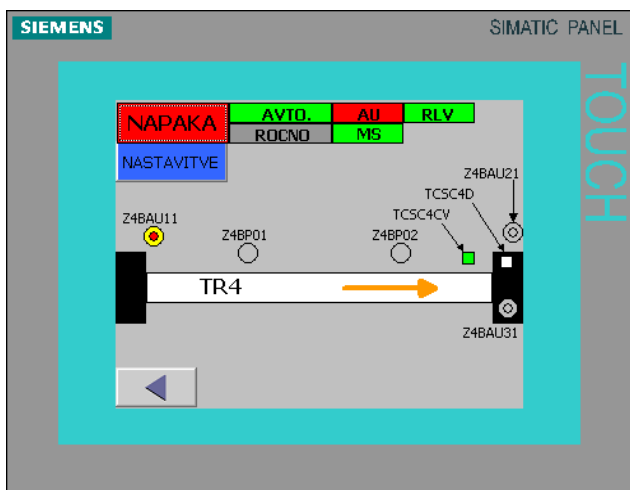
Slika 7: Osnovni zaslon našega nadzornega sistema

Zaradi uporabe 6-palčne HMI-naprave je zaslon precej majhen. Posledica tega je zelo strnjen sinoptični prikaz na glavnem zaslonu. Prikazani so le bistveni elementi za vodenje sistema. Puščice na transporterjih podajajo informacijo o delovanju pogona in smeri gibanja posameznega transporterja. Če pogon deluje (ni nujno, da se transporter giblje), je puščica obarvana zeleno, sicer pa je obarvana oranžno. Ta informacija je zajeta preko spremenljivke za vodenje frekvenčnega pretvornika. V primeru, da je puščica obarvana zeleno in je transporter pri miru, lahko sklepamo, da je prišlo do mehanske okvare na transporterju (npr. pretrganje verige zaradi prevelike obremenitve). Ob vsakem transporterju je napisana tudi trenutna hitrost v metrih na minuto, ki je pridobljena s pomočjo informacij enkoderjev in induktivnih tipal. Za transporter 1 in transporter 2 imamo izpisano tudi razliko pulzov enkoderjev, ki je ključna za pravilno regulacijo hitrosti obeh transporterjev. Podatki na zaslonu se osvežujejo vsakih 500 ms.

S klikom na gumb posameznega transporterja, ki je označen s številko cone in številko transporterja, pridemo na zaslon posameznega transporterja, na katerem so vidna stanja pripadajočih senzorjev in tipk stop v sili ter tipk produkcijski stop, ki so ob posameznem

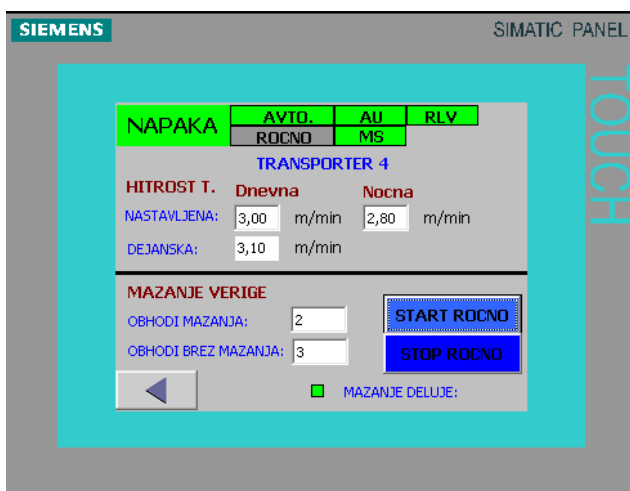
transporterju. Gumb »Rezim delovanja« nam odpre zaslon za vodenje transporterjev za dnevno in nočno izmeno. Če kliknemo na gumb »Nastavitve«, pa se odpre nastavitveni zaslon.

Vsak transporter ima torej svoj zaslon, preko katerega ga krmilimo in opazujemo dogajanje v povezavi z njim (slika 8). Zaslone za vse štiri transporterje imajo enako vsebino.



Slika 8: Zaslon transporterja 4

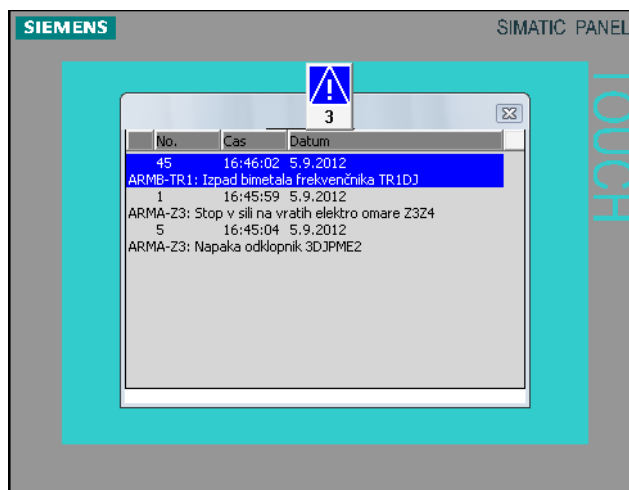
S klikom na gumb »Nastavitve«, ki je obarvan z modro barvo, pridemo na zaslon za krmiljenje transporterja (slika 9). Na krmilnem zaslonu lahko vpisujemo hitrosti za posamezno izmeno, ki so dogovorjene in znotraj dovoljenih meja za posamezen transporter. Sam program prepreči vpisovanje hitrosti, ki so zunaj dovoljenih meja.



Slika 9: Nastavitveni zaslon transporterja 4

V spodnjem delu zaslona imamo tudi vnosna polja za krmiljenje mazanja verige, v katera vpišemo, koliko krogov bo mazanje verige vključeno in koliko krogov bo mazanje verige izključeno.

Za prikazovanje alarmov uporabljamo posebna okna (slika 10), ki se na zaslonu pojavijo le ob sprožitvi enega od alarmov (ali več). Na njih lahko prikažemo več informacij o alarmu (čas, datum, številka alarma, ...). Na zaslonu je izpisano ime tekoče napake, čas in datum njenega nastanka in številka napake. Dokler napake ne odpravimo oziroma je po potrebi ne potrdimo, napaka z zaslona ne izgine.



Slika 10: Zaslon za prikaz alarmov

3 Zaključek

Namen projekta je bila izdelava nadzornega sistema HMI za štiri talne transporterje v programskem okolju WinCC flexible 2008. Nadzorni sistem je zasnovan tako, da operaterju omogoča hiter pregled nad dogajanjem v sistemu in enostavno krmiljenje preko HMI-enote v operaterskem pultu LOP. Odkrivanje napak na sistemu je preprosto, saj se ob vsaki napaki sproži alarm, ki transporter zaustavi in izpiše se opis napake ter čas pojavitve napake. Naprava HMI je torej programirana tako, da operaterju omogoča čim bolj preprosto upravljanje. Sam nadzorni sistem je napisan v treh jezikih (slovenščini, angleščini in francoščini), kar omogoča uporabo tudi ljudem, ki ne razumejo slovenskega jezika.

4 Literatura

- [1] Fakulteta za elektrotehniko, Študijsko gradivo pri predmetu Računalniško vodenje procesov (26.2.2013),
<http://msc.fe.unilj.si/StraniPredmetov.asp?predmet=19>
- [2] Spletna stran Siemens Automation, Multi Panels series 170 - SIMATIC MP 177 (26.2.2013),
<http://www.automation.siemens.com/mcms/human-machine-interface/en/operator-interfaces/multi-panels/series-170/pages/default.aspx>
- [3] Spletna stran Siemens Automation Manuals on WinCC flex (27.2.2013),
<http://www.automation.siemens.com/WW/forum/guests/PostShow.aspx?PageIndex=1&PostID=7892&Language=en>
- [4] Kurose and Ross, "Computer Networks: a top-down approach", Addison Wesley, 2009
- [5] G. Mušič, Računalniško vodenje procesov, Praktikum, Založba FE in FRI, Ljubljana, 2009