

PLC (Programmable Logic Controller) -programming

Veel vaar- en voertuigen maar ook heel veel andere apparaten worden bestuurd door PLC's. Kenmerkend is daarbij een eenvoudig maar effectief besturingsconcept, dat zich uit in een vaste cyclische opeenvolging van de fasen *input*, *sweep* (*verwerking*) en *output* binnen een gegarandeerde responsetijd. De sweep bevat alle besturingslogica. Het uiterlijk van PLC's varieert van dure rack-mount standaard-units tot eenvoudige single-board of zelfs single-chip uitvoeringen, echter het beschreven besturingsconcept is de basis van elke PLC.

PLC-programming verschilt radicaal van imperatief programmeren met *for*, *while* en *if...else* statements zoals gebruikelijk in C(++), Python en Java. Er is sprake van een vaste evaluatievolgorde zonder interne lussen waarbij altijd alle statements worden doorlopen, ook al lijken sommige conditioneel. Deze aanpak maakt zowel realtime on-site debugging als de genoemde gegarandeerde responsetijd mogelijk. Daarnaast heeft de code zo min mogelijk interne state, maar reageert vrijwel instantaan op de externe state van het bestuurd systeem. Dit is cruciaal voor de veiligheid.

PLC's worden geprogrammeerd in een veelheid aan special-purpose talen waarvan vele de suggestie wekken dat het om hard-wired logic gaat. Deze werkwijze is bruikbaar voor eenvoudige toepassingen en vertrouwd voor electrotechnici. Bij complexere toepassingen is representatie als een "echt" computerprogramma veel effectiever en zijn de typische vaardigheden van een TI'er onontbeerlijk. Daarnaast spaart het bij zulke toepassingen veel tijd en geld, gebruik te maken van een "digital twin" van het te besturen systeem, eenvoudigweg geschreven in dezelfde taal als de besturing zelf. Uit de broncode van het betreffende computerprogramma kan indien nodig een representatie voor een willekeurige PLC worden gegenereerd, van ladder-logic tot structured text. Veel moderne PLC's kunnen echter ook rechtstreeks zulke "echte" programmacode draaien.

Met name bij PLC's die voer- en vaartuigen besturen zijn numerieke integratie (van versnelling via snelheid naar verplaatsing), en numeriek differentiatie (van verplaatsing naar snelheid naar versnelling) van belang. De gehanteerde tijdstap Δt is hierbij gelijk aan de cyclustijd en mag binnen zekere grenzen variëren.

Uit het "Opleidingsprofiel Technische Informatica" d.d. Juni 2022:

"Gevraagde technieken in het werkveld van een technisch informaticus zijn onder meer Embedded programmeren, robotica, netwerken en PLC-programmeren, ..."

Het voorstel is met een (keuze- of profiel-)vak recht te doen aan deze vraag.

Jacques de Hooge
11 November 2022