SIL en PL in de Machinebouw

kader
Consultancy & Interim

Bekijk het op
kader.nl

Bekijk het op
kader.nl

Inleiding

In feite zegt een machinebouwer tegen de gebruiker van een nieuwe machine: 'Je kunt gewoon door dat deurtje lopen, mijn hardware en mijn software zorgen ervoor dat je niet geraakt wordt door die robot.'

Veiligheidsfuncties zorgen er bijvoorbeeld voor dat een robot stopt als de deur wordt geopend, of dat een vergrendelde deur pas wordt vrijgegeven nadat de machine tot stilstand is gekomen. De gebruiker van een machine mag dan ook een behoorlijke prestatie (of integriteit) van deze veiligheidsfuncties verwachten.

De machineverordening eist dat een fout in de besturing (hardware of software) niet tot een gevaarlijke situatie leidt. We gebruiken de normen EN ISO 13849 (PL) en EN IEC 62061 (SIL) om te kunnen voldoen aan deze wettelijke eisen.

Het gemiddelde besturingssysteem van een machine bevat steeds meer veiligheidsgerelateerde onderdelen. Waar in het verleden uit het elektrische schema's nog wel af te leiden was hoe de veiligheidsfuncties werkten, zien we in nieuwe machines vaak een veiligheids-PLC met een netwerkkabel naar een aantal frequentieregelaars. Dat zorgt voor veel technische mogelijkheden maar zowel de normen als de machineverordening eisen dat veiligheidsgerelateerde software wordt gevalideerd met behulp van een traceerbaar ontwerptraject.

Pas na een zorgvuldig, gedocumenteerd en traceerbaar ontwerptraject, met verificatie en validatie, kan een machinebouwer er zeker van zijn dat de veiligheidsfuncties het juiste niveau hebben.

In deze brochure worden de belangrijkste begrippen uit de SIL- en de PL-norm samengevat. Ook is er aandacht voor een gestructureerd ontwerptraject met behulp van het V-model.

Contact
Heb je na het lezen van de informatie nog vragen, of wil je een offerte aanvragen? Neem dan gerust contact met ons op.

Telefoonnummer 030 243 64 64
kader.nl | sales@kader.nl

Wij helpen je graag!

Veiligheidsfuncties

Risicobeoordeling
De risicobeoordeling is een belangrijk instrument om tot een veilige machine te komen. Het is noodzakelijk om hiermee al vroeg in het ontwerptraject te starten. Alleen op die manier kunnen de juiste stappen worden doorlopen om risico's te reduceren. Het komt helaas vaak voor dat de risicobeoordeling pas aan het eind wordt opgesteld, soms zelfs in de vorm van een checklist. Effectieve maatregelen (in het machineontwerp) kunnen dan niet meer genomen worden.

Grenzen
Leg eerst de grenzen van de machine vast, met voldoende detail: bedoeld gebruik, redelijkerwijs voorzienbaar verkeerd gebruik, fysieke grenzen van de machine, gebruikers, bedrijfsmodi, omgevingsomstandigheden, energiebronnen, service-intervallen, toegankelijkheid, bedien- en onderhoudslocaties, afmetingen, massa's, cyclustijden, eigenschappen van materialen in de machine enzovoorts.

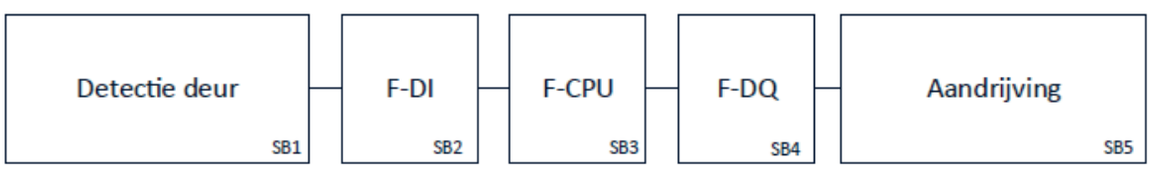
Lijst met gevaren
Identificeer 'alle' gevaren in de machine, zonder rekening te houden met eventuele beveiligingen. Ga nog geen oplossingen bedenken! Beschouw (op een gestructureerde manier) alle fasen in de levenscyclus van de machine en alle taken die aan de machine worden uitgevoerd. Identificeer gevaren in de vorm van scenario's.

Inschatten van de risico's
Als een gevaar weinig letsel kan veroorzaken of er is nauwelijks blootstelling aan dat gevaar zal het risico lager zijn. Schat voor elk gevaar in hoe groot het risico is. Gebruik een methode zoals risicograaf of Kinney&Wiruth.

Risico's reduceren in 3 stappen
1. Verbeter het ontwerp, zorg dat het gevaar er niet meer is.
2. Voeg een beveiliging toe, het gevaar kan niet worden weggenomen maar de blootstelling wordt verminderd.
3. Verstrek informatie, er is nog een gevaar met blootstelling, veilig werken vereist bijvoorbeeld instructie en persoonlijke beschermingsmiddelen. Soms kan een geluidssignaal als waarschuwing voor de start van een beweging een laatste stap van risicoreductie zijn.

EN ISO 12100
Safety of machinery – General principles for design – Risk assessment and risk reduction

Functionele veiligheid
Een vaste afscherming is een voorbeeld van een beveiligingsmaatregel. Hoewel er eisen zijn aan vaste afschermingen heeft dit niets met SIL of PL te maken. Pas als een beveiliging uitgevoerd wordt door het besturingssysteem, spreken we van een veiligheidsfunctie.



Voorbeeld blokdiagram subsystemen

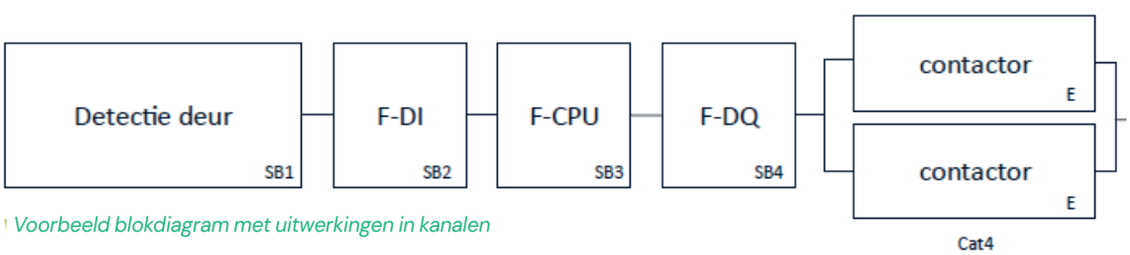
Veiligheidsfunctie
Een functie die nodig is om een veilige toestand voor de machine te bereiken of te behouden, met betrekking tot een specifieke (gevaarlijke) gebeurtenis. Bijvoorbeeld: als de deur wordt geopend moet de aandrijving stoppen (zie blokschema), de brug mag alleen omhoog als de bomen dicht zijn, de cilinder wordt drukloos als het lichtscherm wordt doorbroken, de vergrendelde deur wordt pas vrijgegeven nadat de machine tot stilstand is gekomen. Veiligheidsfuncties worden uitgevoerd door het (veiligheidsgerelateerde deel van het) besturingssysteem.

Benodigd level
Afhankelijk van het risico krijgt elke veiligheidsfunctie een benodigd Performance level PL (a-e) of een benodigd SIL Level (1-3).

Subsysteem
De veiligheidsfunctie wordt opgedeeld in subsystemen (sensor – logic – actuator) waarbij elk subsysteem een architectuur moet hebben die robuust genoeg is voor het te behalen level. Daarnaast moet de opgetelde faalkans van de subsystemen laag genoeg zijn. Onafhankelijk van de berekening moet ook aandacht besteed worden aan kwalitatieve aspecten en vakmanschap.

Kanaal
Een subsysteem kan bestaan uit één (enkelvoudig) of twee kanalen (redundant).

Element /blok
Een kanaal bevat één of meerdere elementen (deze worden soms ook blok genoemd). In het voorbeeld hierboven wordt de aandrijving gestopt door een 2-kanaals subsysteem met een contactor als element (E) in elk kanaal.



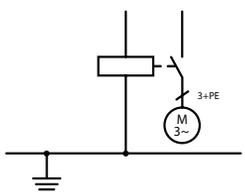
Voorbeeld blokdiagram met uitwerkingen in kanalen

Performance level PL
EN ISO 13849:2023
Safety of machinery - Safety-related parts of control systems

Safety integrity level SIL
EN IEC 62061:2021
Safety of machinery - Functional safety of safety related control systems

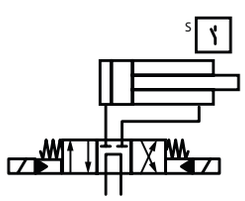
Architectuur en faalkans

Robuuste architectuur, per subsysteem
Hoewel een enkelvoudig systeem een hele lage faalkans kan hebben is het niet mogelijk om met deze architectuur een hoog level te bereiken. Voor de lagere levels (PL a, PL b, PL c of SIL 1) is een 1-kanaals architectuur meestal voldoende. De hogere levels (PL d, PL e, SIL2 of SIL3) worden doorgaans met een 2-kanaals architectuur met diagnose uitgevoerd.



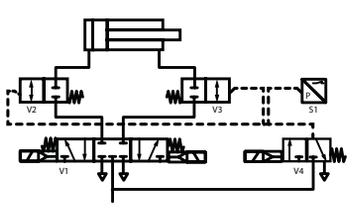
Enkelvoudig systeem

Een gevaarlijke fout, zoals het verkleven van de motorcontactor, kan leiden tot het verlies van de veiligheidsfunctie. De kwaliteit van dit subsysteem is volledig afhankelijk van de kwaliteit van de enkele motorcontactor. Gebruik altijd beproefde componenten.



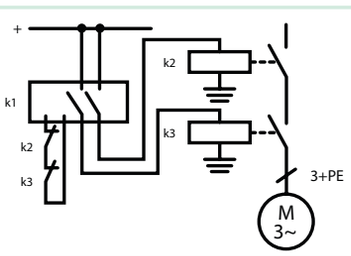
Getest enkelvouwdig systeem

De correcte werking van het hydraulische stuurventiel wordt indirect teruggekoppeld door sensor S. Bij falen van het stuurventiel kan bijvoorbeeld het hydraulische aggregaat worden gestopt om de cilinder alsnog veilig tot stilstand te brengen. Let op: er zijn strikte eisen aan de testfrequentie en de foutreactie, na diagnose van een fout moet de machine in een veilige toestand komen.



Redundant systeem met lage DC waarde

Het stuurventiel met gesloten middenstand kan de cilinder stilzetten (kanaal 1). De blokkeerventielen, aangestuurd door een extra stuurventiel, kunnen dat ook (kanaal 2). Tenminste één van beide kanalen moet functioneren om de cilinder te stoppen. Het is niet ondenkbaar dat een fout in een kanaal pas na een tijdje wordt gedetecteerd.



Redundant systeem met hoge DC waarde

Door de mechanisch gedwongen verbreekcontacten van de contactors K2 en K3 terug te koppelen naar de veiligheidsbesturing worden fouten met hoge zekerheid gedetecteerd. De machine kan, na het falen van één kanaal, niet meer opstarten.

Categorie

Onderverdeling van de gebruikte architectuur gebaseerd op 1 of 2 kanalen, diagnose en reactie op fouten. Wordt alleen gebruikt in de PL-norm.

Hardware fault tolerance (HFT)

Tegen hoeveel fouten is het subsysteem bestand? Een 1-kanaals subsysteem is niet bestand tegen fouten (HFT0). Een 2-kanaals subsysteem is bestand is tegen één fout (HFT1).

Safe failure fraction (SFF)

Het percentage veilige fouten (waarbij bijvoorbeeld de machine veilig stopt als gevolg van de fout) aangevuld met het percentage gevaarlijke fouten dat gedetecteerd wordt (zodat de machine in een veilige toestand gebracht kan worden).

Maximum SIL that can be claimed

Geeft voor elk subsysteem aan tot welk SIL niveau deze architectuur mag worden toegepast, wordt via een tabel bepaald op basis van HFT en SFF. In het verleden werd hiervoor de variabele SILCL gebruikt. Hoewel vervallen is deze term nog volop in gebruik.

Lage faalkans
Per subsysteem wordt de kans op toevallig gevaarlijk falen (Random Hardware Failure) bepaald of berekend. De som van de faalkansen van de subsystemen vormen de faalkans van de veiligheidsfunctie. Deze faalkans moet laag genoeg zijn voor het benodigde level.

Gevaarlijk falen Met een component dat gevaarlijk faalt is een machine niet meer naar de veilige toestand te brengen	Een contactor die blijft plakken (kan niet uitschakelen) is gevaarlijk gefaald omdat deze een motor niet kan stoppen. Een contactor met een doorgebrande spoel (kan niet inschakelen) is veilig gefaald omdat deze een motor niet kan starten.	
PFH De frequentie waarmee het subsysteem er niet in slaagt de veiligheidsfunctie uit te voeren (per uur).	De PFH (Probability of dangerous Failure per Hour) van de veiligheidsfunctie wordt berekend door de PFH-waarden van alle subsystemen op te tellen. Als de PFH-waarde niet door de leverancier wordt opgegeven, moet deze bepaald worden met behulp van de andere variabelen in deze tabel. (De term PFHD wordt ook gebruikt en heeft dezelfde betekenis als PFH).	
$MTTF_D$ Gemiddelde tijd, in jaren, tot het gevaarlijk falen van een component.	**λ_D** Faalfrequentie, frequentie waarmee een component gevaarlijk faalt (per uur).	$MTTF_D$ (Mean Time To Failure dangerously) en λ_D zijn gedurende de te bepalen levensduur (een bepaalde tijd of aantal schakelingen) constant. Dan geldt $MTTF_D$ is $1/\lambda_D$. Aan het einde van deze levensduur moeten componenten preventief worden vervangen omdat λ_D daarna snel zal stijgen en $MTTF_D$ zal dalen.
B_{10D} Statistisch aantal schakelingen (voor elektromechanische componenten) waarna 10% van de componenten gevaarlijk heeft gefaald.	**nop** of **c** Aantal keer schakelen (voor dit component) per jaar (nop) of per uur (c).	Met de B_{10D} en de nop (number of operations) of c-waarde is met een eenvoudige formule de $MTTF_D$ of λ_D waarde te berekenen.
RDF het percentage fouten dat gevaarlijk is ten opzichte van alle fouten.	Als bijvoorbeeld een contactor een B_{10} waarde van 1.000.000 heeft met een RDF (Ratio of Dangerous Failure) van 73% dan is: $B_{10D} = B_{10}/RFD = 1.000.000 / 0,73 = 1.369.863$	
DC Het percentage gevaarlijke fouten dat door de diagnose wordt gedetecteerd.	Een redundant subsysteem is pas zinvol als fouten worden gedetecteerd. Afhankelijk van de gebruikte techniek van foutdetectie wordt een DC (Diagnostic Coverage) waarde toegekend aan elementen. De combinatie van faalkans ($MTTF_D$ of λ_D), architectuur en diagnose (DC) wordt gebruikt in de bepaling van de PFH-waarde van het subsysteem.	

Goed vakmanschap

Als een contactor faalt omdat er te veel stroom loopt of een schakelaar faalt omdat deze niet bestand is tegen de omgevingsomstandigheden, dan is dat geen toevallig falen maar een direct gevolg van een fout ontwerp. Deze fouten zijn dus systematisch. Het vervangen van de kapotte schakelaar door een (identieke) nieuwe schakelaar lost het probleem niet op. Bij het berekenen van de faalkans is altijd het uitgangpunt dat de componenten precies op de juiste manier worden toegepast, zoals dit door de leverancier in de datasheet is vermeld.

Pas basis- en beproefde veiligheidsprincipes toe zoals:
Kies geschikte materialen en componenten; gebruik datasheets en manuals; installeer de voorgeschreven beveiligingen tegen te hoge stroom, druk, temperatuur etc.; zorg voor de juiste lucht en oliekwaliteit bij hydraulische en pneumatische systemen; houd rekening met omgevingscondities.

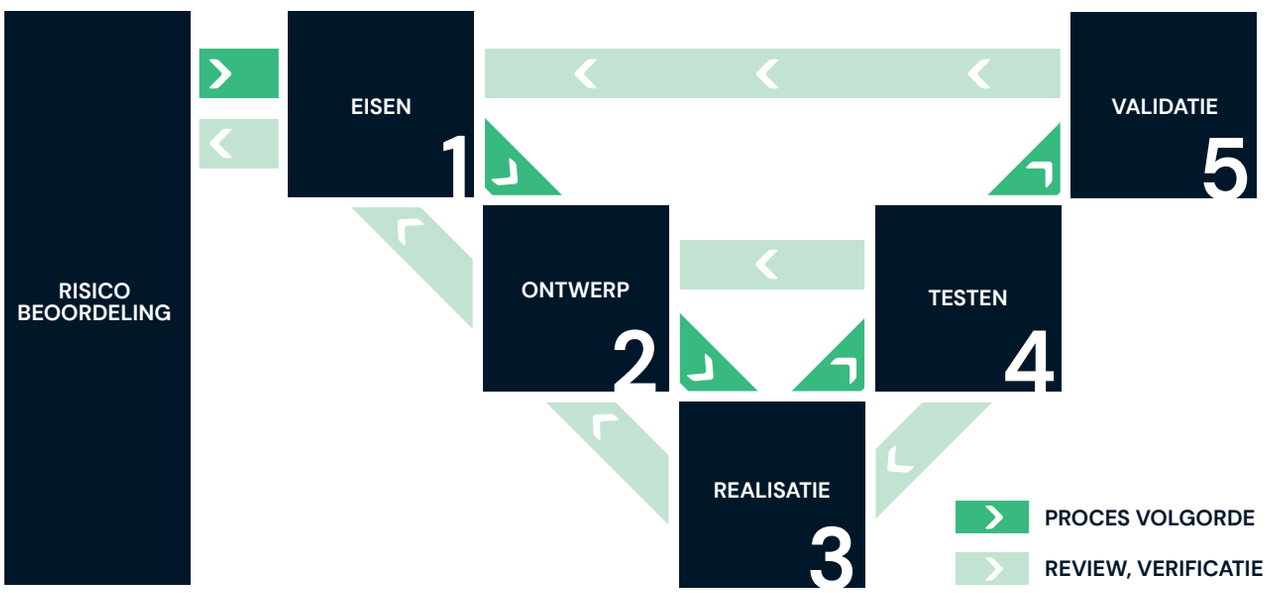
CCF Bepaal of voldoende aandacht is besteed aan het voorkomen van Common Cause fouten met behulp van een checklist.	Een Common Cause fout is één fout die beide kanalen laat falen, zoals een overbelasting waardoor contactors tegelijk blijven kleven. Voor PL moet een bepaald aantal punten worden gescoord in de checklist, voor SIL volgt uit de checklist nog een CCF-factor β die gebruikt wordt in de PFH-berekening.

De gebruiksaanwijzing

Dit is een omvangrijk document, of eigenlijk een serie van documenten, dat alle informatie moet bevatten die de gebruiker nodig heeft om de machine goed en veilig op zijn plaats te krijgen, te gebruiken, te onderhouden, te repareren en te demonteren. Dus inclusief de operator handleiding, de service handleiding, de installatie instructies, de elektrische schema's, transportgegevens enzovoorts. In veel normen is terug te vinden wat er specifiek voor dat onderwerp in de gebruiksaanwijzing moet worden opgenomen.

Systematische werkwijze
Iedereen mag fouten maken, en dat gebeurt ook. Maar als de machine in bedrijf gaat, mogen de gebruikers wél een zekere performance of integriteit van de veiligheidsfuncties verwachten. Op dat moment mogen er geen fouten meer in zitten, want dat zou tot een ongeval kunnen leiden.

Binnen deze werkwijze is het dus belangrijk dat een ontwerper pas gaat ontwerpen als er een gereviewd eisenpakket ligt en dat de tekenaar pas gaat tekenen als er een gereviewd ontwerp ligt. De tijd en energie die het kost om deze formele werkwijze toe te passen, wordt uiteindelijk terugverdiend door minder issues tijdens de bouw en inbedrijfname. Maar het leidt vooral tot betere en veilige machines.



1. **Eisen**
   Beschrijf waar de veiligheidsfuncties aan moeten voldoen (SRS).

2. **Ontwerp**
   Leg vast met welke componenten en architecturen aan de eisen kan worden voldaan (HDS – SDS).

3. **Realisatie**
   Pas de ontwerpen toe en maak schema's, software en gebruiksaanwijzing.

4. **Testen**
   Controleer of de subsystemen en de veiligheidsfuncties doen wat er in het ontwerp is vastgelegd.

5. **Validatie**
   Stel vast dat elke veiligheidsfunctie de prestatie levert die vereist is in de SRS.

SRS (SaRS) Safety Requirements Specification
Beschrijf de veiligheidsfuncties en de prestatie die deze moeten leveren (SIL of PL -niveau) inclusief reactiesnelheid, omgevingsomstandigheden, bedrijfsmodi en eventuele uitzonderingen.

HDS (SaHDS) Hardware Design Specification
Werk de veiligheidsfuncties uit in blokschema's en beschrijf elk subsysteem en de toegepaste componenten. Toon aan dat met de gekozen oplossing het benodigde niveau kan worden behaald en dat de componenten bestand zijn tegen de omgevingsomstandigheden zoals gespecificeerd in de eisen.

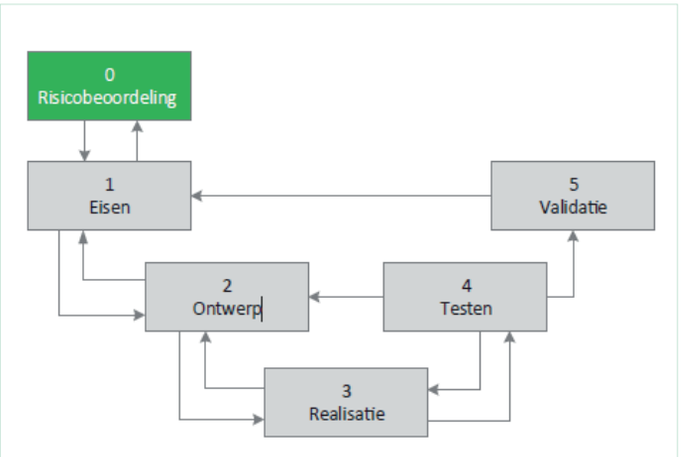
SDS (SaSDS) Software Design Specification
Ontwerp de veiligheidsgerelateerde software. Beschrijf de architecturen, testen en toe te passen functieblokken, zodat de programmeur een leesbaar, begrijpelijk en toetsbaar programma kan maken.

Verificatie en validatie
Bij de validatie, helemaal aan het einde van het traject, wordt vastgesteld dat alle veiligheidsfuncties inclusief alle eisen (uit de SRS) ook daadwerkelijk zijn geïmplementeerd. Voordat een stap wordt afgesloten moet (door review, analyse of testen) worden geverifieerd of deze nog in lijn is met eerdere stappen.

Functional safety plan
Dit plan beschrijft hoe functionele veiligheid wordt aangepakt. Welke stappen worden doorlopen? Hoe vindt verificatie en validatie plaats? Welke personen zijn hiervoor verantwoordelijk en hoe wordt alles vastgelegd.

V-model
Systematische werkwijze

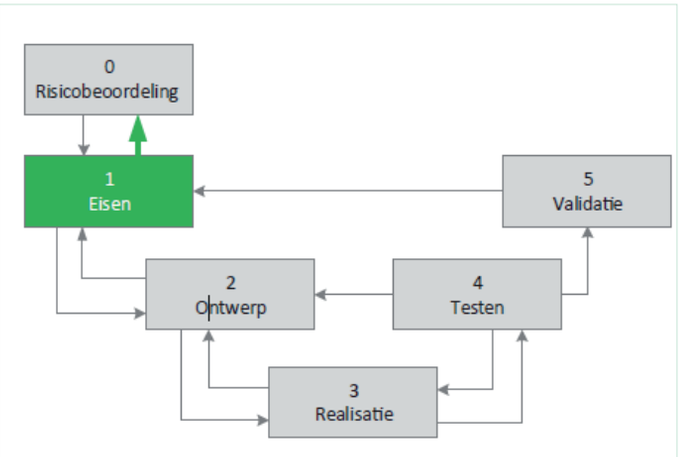
Stap 0: Risicobeoordeling



De eerste stap is de risicobeoordeling. Deze stap valt eigenlijk buiten het V-model omdat het nog niet om functionele veiligheid gaat. Pas na het uitvoeren van een risicobeoordeling, worden veiligheidsfuncties als risicoreducerende maatregel gespecificeerd. De eerste risicobeoordeling wordt vroeg in het ontwerp uitgevoerd, maar meestal zijn er meerdere iteraties en worden er nog zaken geactualiseerd. Probeer de veiligheidsfuncties in een vroeg stadium 'compleet' te hebben en zorg voor een goed wijzigingsprotocol als er nog zaken veranderen.

Actie
Voer de risicobeoordeling uit
- Bepaal de grenzen van de machine.
- Maak een lijst van gevaren.
- Schat per gevaar het risico in.
- Definieer risico-reducerende maatregelen.
- Identificeer de veiligheidsfuncties.

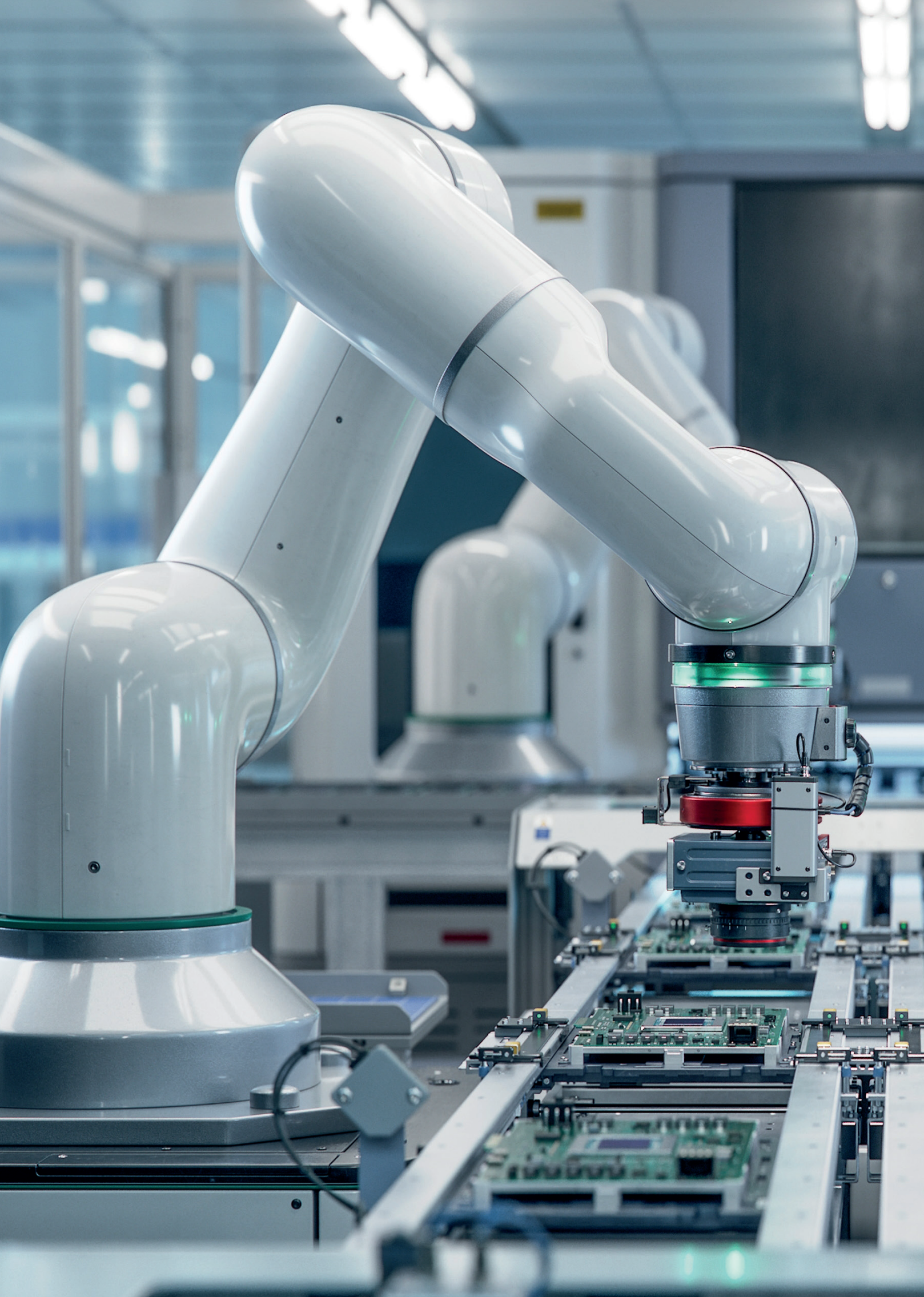
Stap 1: Eisen

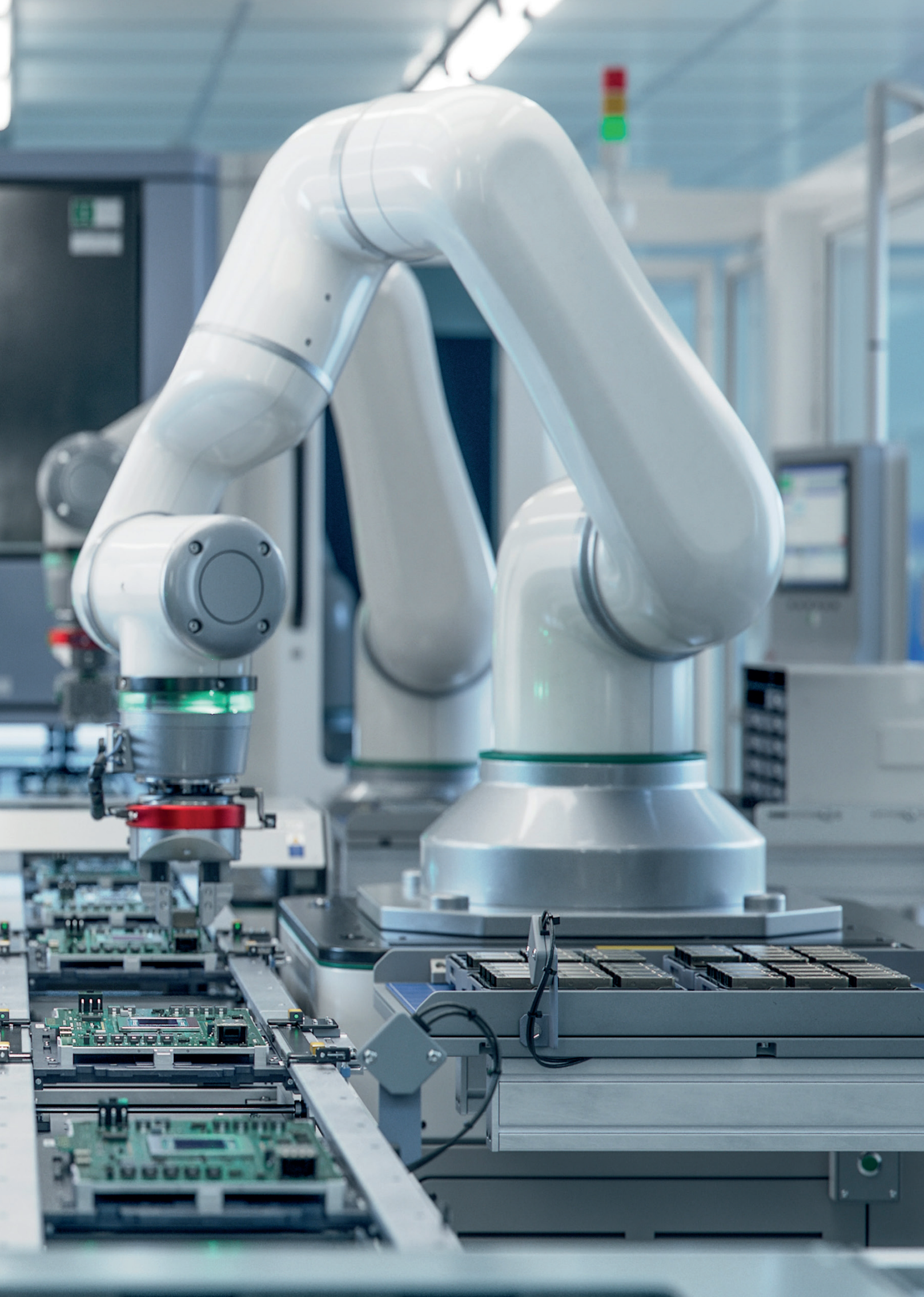


Nadat de veiligheidsfuncties als risicoreducerende maatregel uit de risicobeoordeling volgen, is het belangrijk om de eisen aan deze functies op te stellen. Het gaat nadrukkelijk om de functie en nog niet om de componenten of de gekozen oplossing. De specificatie van de eisen heet: Safety Requirements Specification (SRS). De eisen vormen het uitgangspunt van de diverse ontwerpen, maar worden ook aan het einde van het traject gebruikt als tijdens de validatie wordt aangetoond dat elke veiligheidsfunctie voldoet aan de eisen.

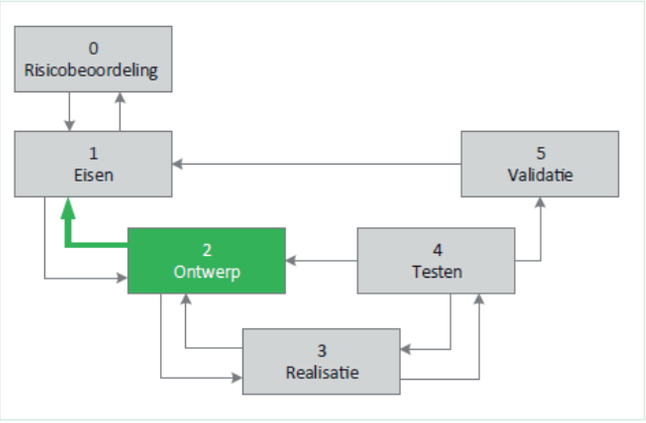
Actie
Stel een SRS-document op waarin de eisen aan de veiligheidsfuncties worden gespecificeerd, Beschrijf voor elke veiligheidsfunctie o.a.:
- Identificatie/naam.
- Link naar de risicobeoordeling.
- Functionaliteit en aanspreekfrequentie.
- Benodigd PL- of SIL-level.
- Omgevings- en gebruiksomstandigheden.
- Gebruiksmodi en uitzonderingen.

Verifieer
Zijn alle veiligheidsfuncties die uit de risicobeoordeling volgen volledig gespecificeerd met het oog op het risico dat door de veiligheidsfunctie gereduceerd wordt?
- Passen de gespecificeerde omgevingseisen bij de grensbeschrijving?
- Is de specificatie volledig en eenduidig?





Stap 2: Ontwerp



Nadat het SRS-document is vrijgegeven, kunnen de ontwerpers aan de slag. De veiligheidsfuncties worden uitgewerkt in blokschema's (subsystemen, kanalen en elementen) en componenten worden geselecteerd.

Meteen daarna wordt het software ontwerp gemaakt. Voor software is een eigen V-model gedefinieerd. Deze ontwerpdocumenten worden gebruikt tijdens de realisatie, dus bij het tekenen van de schakelschema's en het programmeren van de software, maar ook tijdens het testen als aangetoond wordt dat de machine werkt zoals in het ontwerp is beschreven.

Actie

Ontwerp de veiligheidsfuncties:

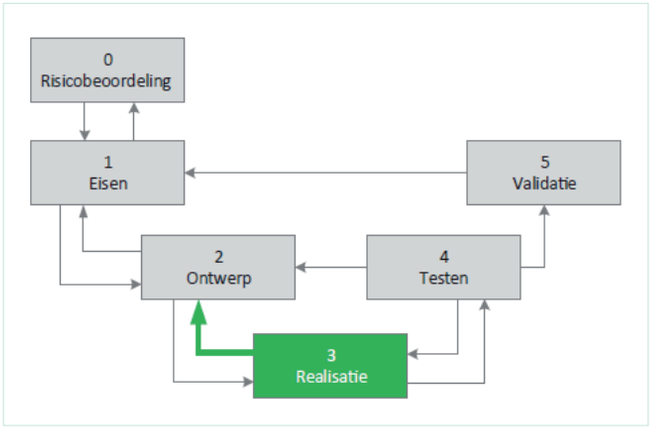
- Decompositie in subsystemen (Sensor – Logic – Actuator).
- Componenten, architectuur, diagnose.
- Ontwerp de veiligheidsgerelateerde software. (*).

Verifieer

Kwantitatieve aspecten:

- Is de gekozen architectuur robuust genoeg (categorie/ max SIL, SILCL)?
- Is de berekende faalkans laag genoeg (PFH)?
- kwalitatieve aspecten:
  Zijn componenten onderling compatibel?
- Zijn componenten bestand tegen de omgevingsomstandigheden?
- Volledigheid en eenduidigheid, zijn alle veiligheidsfuncties opgenomen? Kan met het software ontwerp een leesbaar begrijpelijk en toetsbaar programma woorden geschreven? (*)

Stap 3: Realisatie



Als de ontwerpdocumenten zijn vrijgegeven, kunnen de detailtekeningen en schema's worden gemaakt en wordt de software geschreven. Dit is ook het moment om de gebruiksaanwijzing op te stellen. Na controle van deze documenten wordt de machine gebouwd.

Actie

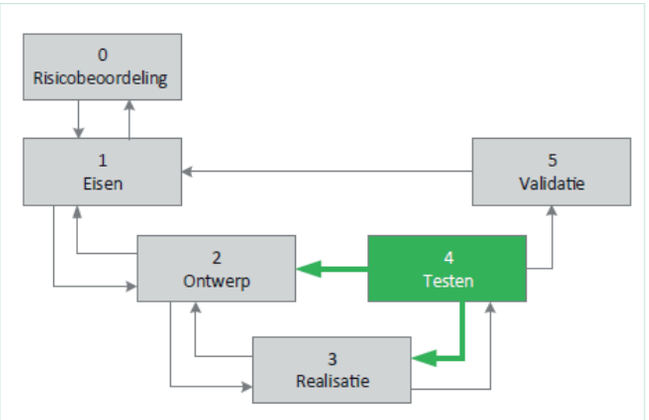
- Maak tekeningen en schema's (elektrisch, hydraulisch, pneumatisch).
- Schrijf de gebruiksaanwijzingen.
- Programmeer veiligheidsgerelateerde software (*).
- Bouw daarna de machine.

Verifieer

- Zijn alle componenten uit de ontwerpdocumenten correct in de tekeningen opgenomen?
- Zijn er (geen) componenten toegevoegd?
- Zijn de componenten toegepast volgens de gebruiksaanwijzing?
- Is de diagnose op de juiste manier verwerkt?
- Is de hardware volgens tekeningen gerealiseerd?
- Code review: matcht het programma met het ontwerp? (*)

** software heeft een eigen V-model*

Stap 4: Testen



Als de linkerzijde van het V-model juist is doorlopen (inclusief de verificaties), dan zal de kans op fouten in de machine de nul naderen. Uiteraard zal dit tijdens de testfase worden bevestigd. In het V-model zijn de testen altijd gerelateerd aan het ontwerp. Test of de machine werkt zoals beschreven in het ontwerp. Bij een goed uitgevoerd V-model is het een zeldzaamheid dat er fouten worden ontdekt tijdens het testen. Onderzoek de oorzaak van deze fouten en verbeter daarmee het ontwerp en review traject zodat bij een volgend project deze fouten al aan de linkerkant van de V worden opgemerkt.

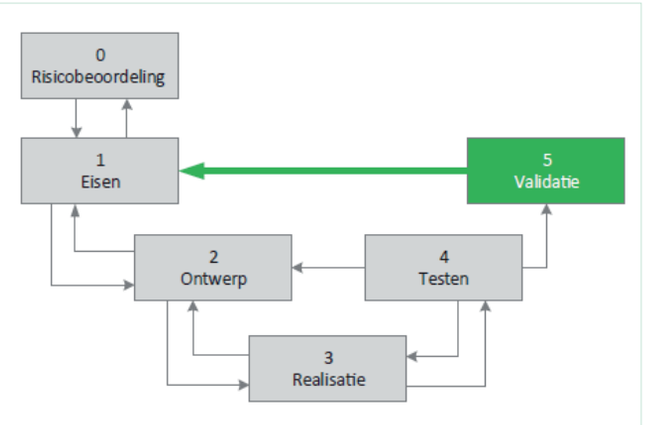
Actie

- Test de I/O van de veiligheidsPLC, inclusief instellingen en diagnose.
- Test de redundantie van de subsystemen, introduceer fouten, test de diagnose.
- Test de veiligheidsfuncties functioneel.

Verifieer

- Functioneert het veiligheidsgerelateerde deel van het besturingssysteem, zoals beschreven in het ontwerp?
- Functioneren alle componenten zoals voorzien?

Stap 5: Validatie



Als een veiligheidsfunctie is gevalideerd, kan de gebruiker van de machine ervan op aan dat het risico in de machine op de juiste manier is gereduceerd en dat dit deel van de machine veilig is. De operator kan bijvoorbeeld door het lichtscherm lopen en het besturingssysteem (hard- en software) zorgt ervoor dat de machine geen gevaarlijke bewegingen meer uitvoert. De machinebouwer steekt hiervoor spreekwoordelijk zijn/haar hand in het vuur! Uiteraard wordt deze stap ook formeel vastgelegd.

Actie

- Controleer het ontwerptraject met de bijbehorende documentatie (documenten, verificatie en traceerbaarheid).
- Controleer de testrapporten.

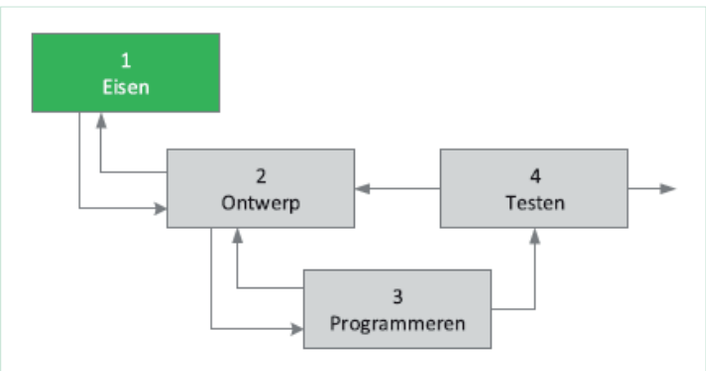
Verifieer

- Zijn alle stappen in het model doorlopen en gedocumenteerd en zijn alle verificaties uitgevoerd?
- Is voor de veiligheidsfunctie voldaan aan de eisen die gespecificeerd zijn?

Software in veiligheidsPLC's

Veiligheidsgerelateerde software
Voor applicatiesoftware in een veiligheidsPLC kan een vereenvoudigd V-model gebruikt worden. De SIL- en PL-normen eisen dat er gebruik gemaakt wordt van dit V-model bij de ontwikkeling van veiligheidsgerelateerde software.

Stap 1: Eisen

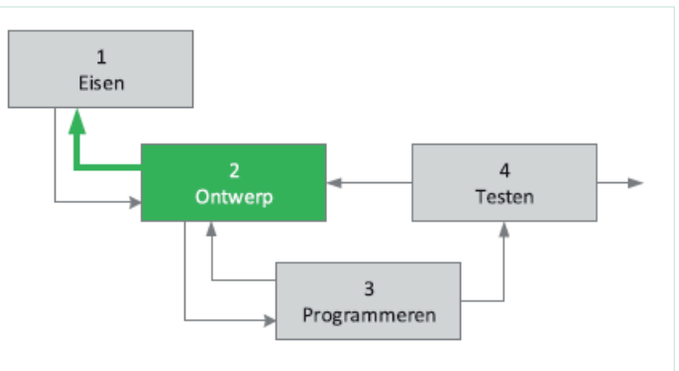


In de praktijk worden de eisen aan software al grotendeels beschreven door specificatie van de veiligheidsfuncties (SRS), het hardware ontwerp (HDS) en de I/O lijst, aangevuld met de coding guidelines. Het eisenpakket voor veiligheidsgerelateerde software wordt dus meestal 'samengesteld' uit verwijzingen naar deze losse documenten.

Actie
Beschrijf:
- Specificatie van de veiligheidsfuncties, met de benodigde SIL- of PL- levels (SRS).
- Omschrijving (of schets) van de architectuur, subsystemen en componenten (HDS).
- Beschrijving van de hardware inclusief PLC en I/O (HDS).
- Diagnose en gedrag van sensoren en actuatoren: hoe komt de machine in een veilige toestand (HDS)?
- Operator interfaces (schakelaars, modeselectie, UI) (SRS+HDS).
- Bedienmodi waarin de veiligheidsfunctie actief is (SRS).
- Eisen aan de reactietijd van de functie (SRS).
- Coding guidelines.

Verifieer
Controleer de verwijzingen naar de SRS en HDS, let op de traceerbaarheid.

Stap 2: Ontwerp

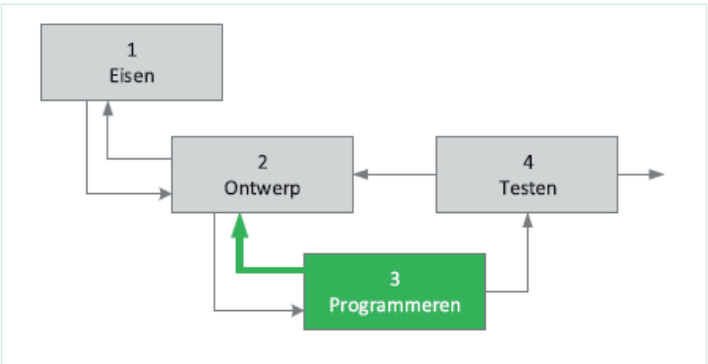


Als de eisen goed zijn vastgelegd, kan de ontwerper een software design document (SDS) opstellen. Dit document beschrijft hoe het programma opgebouwd moet worden. Ook als de programmeertaal simpel is en het programma kort, is het opstellen van een ontwerpdocument belangrijk.

Actie
Beschrijf:
- Logica en architectuur van de veiligheidsfuncties en de I/O.
- Diagnosefuncties op sensoren en actuatoren.
- Functies die de machine in een veilige staat brengen of houden.
- Functies voor detectie van fouten.
- Teststappen/ hoofdlijnen van een testplan.
- Functies met betrekking tot periodiek testen.
- Maatregelen tegen ongeautoriseerd wijzigen.
- Interfaces met de reguliere besturing.
- Responstijden.
- PLC afhankelijke aspecten zoals:
  - omgang met data en variabelen,
  - te gebruiken gevalideerde softwarebibliotheken,
  - programmeertools.

Verifieer
Voer een review uit op de SDS. Let op volledigheid en eenduidigheid.
- Zijn alle onderdelen uit de eisen correct in het ontwerp opgenomen?
- Is het een leesbaar document (ook voor een niet-software specialist)?
- Bevat het alleen relevante zaken?
- Kan de programmeur met dit ontwerp een leesbaar, begrijpelijk en toetsbaar programma schrijven?
- Is het ontwerp terug te traceren naar de eisen?
- Is het testplan compleet en in lijn met het ontwerp?

Stap 3: Programmeren



Nadat het software ontwerp is vrijgegeven wordt de software geschreven. De programmeur kan met de coding guidelines en het software design document een leesbaar, begrijpelijk en toetsbaar programma maken. Als dat niet het geval is, dan is het ontwerp niet goed en moet de programmeur het document niet accepteren. Met behulp van een code review wordt daarna gecontroleerd of het programma precies matcht met wat er in het ontwerp staat.

Actie

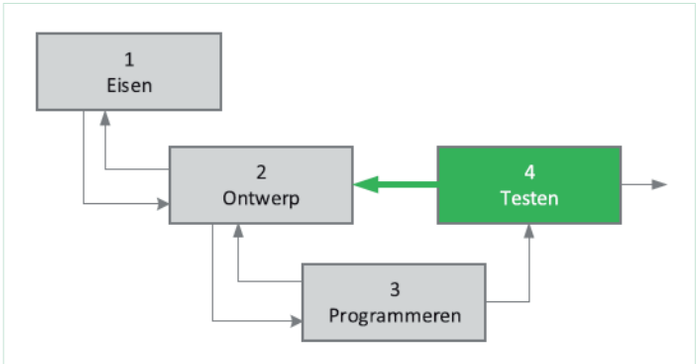
- Schrijf de software volgens de SDS en gebruik daarbij de coding guidelines.
- Focus op een leesbaar, begrijpelijk en toetsbaar programma.

Verifieer

Voer een code review uit:

- Is het programma in lijn met het design?
- Is er niets vergeten?
- Is er niets toegevoegd?
- Coding guidelines.

Stap 4: Testen



Na een correct ontwerptraject en een goede code review nadert de kans op fouten in de software de nul. Na een succesvol uitgevoerde test op basis van een gecontroleerd testplan kan de software worden gevalideerd. Bij een goed uitgevoerd V-model is het een zeldzaamheid dat er fouten worden ontdekt tijdens het testen. Onderzoek de oorzaak van deze fouten en verbeter daarmee ook het ontwerptraject, zodat fouten zoals deze bij een volgend project al aan de linkerkant van de V worden opgemerkt.

Actie

- Voer het testplan uit, rapporteer de bevindingen.
- Doet het programma precies dat wat beschreven is in het design document?

Verifieer

- Eisen aan de reactietijd van de functie (SRS).
- Coding guidelines.

BESTEL HET BOEK:
SIL & PL IN DE MACHINEBOUW

Als machinebouwer of ontwerper is machineveiligheid het belangrijkste onderdeel van je werk. Om de veiligheid te waarborgen zijn vaak veiligheidsfuncties nodig. Het juist berekenen van de toevallige faalkans krijgt hierbij veel aandacht. Maar veiligheid is meer dan dat. Kwaliteit en vakmanschap tijdens het ontwerp van de machine moet niet worden onderschat. Zijn alle componenten juist gekozen? Worden ze goed gemonteerd? En is er gecontroleerd of het materiaal tegen bepaalde omstandigheden kan?

In het boek 'SIL & PL in de machinebouw: een actuele kijk op functionele veiligheid' gaat onze collega Gerald van Engeland in op het belang van vakmanschap en het hanteren van een goed proces bij het waarborgen van de veiligheid in de machinebouw.

Bestel hem hier!

Ook interessant voor jou?

Opleiding
SIL en PL specialist met TÜV certificaat
In deze praktijkgerichte opleiding leer je hoe je machinebesturingen analyseert en optimaliseert volgens de normen voor Performance Level (PL) en Safety Integrity Level (SIL). Behaal na succesvol afronden van het examen (in het Nederlands) het TÜV-Rheinland SIL & PL Specialist, machine safety persoonscertificaat.

Training
Basistraining Performance Level en SISTEMA
Na de training ben je in staat om veiligheidsfuncties op te stellen en het behalen van het SIL level aan te tonen door berekeningen. Door de praktische oefeningen met de tools krijg je een goed begrip van hoe de SIL norm kan worden toegepast.

Training
Software in veiligheids PLC
De trainingen van PLC leveranciers zijn technisch, inhoudelijk en gericht op specifieke software en hardware. Deze trainingen gaan meestal onvoldoende in op de veiligheidsaspecten. Want; Wat vereist de PL of SIL norm? Hoe weet u dat software foutloos is? Is testen voldoende? Heeft software ook een Performance Level? Wat moet er allemaal vastgelegd worden? Hoe zit het met de verantwoordelijkheid?

Over Kader Group

Kader Group verbetert de kwaliteit, veiligheid en duurzaamheid van organisaties, zodat een ieder goed kan zorgen voor mens, milieu en maatschappij. Dat doen we met onze consultancy & interim diensten, digitale oplossingen en opleidingen op de gebieden kwaliteit, veiligheid & gezondheid, cyber security en milieu & duurzaamheid. Onze missie: het creëren van een veilige en duurzame leefomgeving om door te geven aan toekomstige generaties. We zijn daarin pragmatisch, innovatief, persoonlijk en vakkundig.

Kader Group startte in 1994 en bestaat inmiddels uit ruim 200 gedreven medewerkers. Vanuit onder meer 3 business units helpen wij onze opdrachtgevers bij te dragen aan een betere leefomgeving:

- Academy
- Consultancy & Interim
- Digital

Bekijk het op
kader.nl

Contact

Heb je na het lezen van de informatie nog vragen?
Neem dan vooral contact met ons op.

Telefoonnummer 088 - 995 13 20
kader.nl | sales@kader.nl

Wij helpen je graag!

V.0426