

Bionic Robots

- Begleitheft
- Activity booklet
- Manuel d'accompagnement
- Begeleidend boekje
- Cuaderno adjunto
- Folheto



fischertechnik® 

D INHALT

1.	Bionic – Die Natur als Vorbild	S. 2
2.	Voraussetzungen und Einstieg	S. 2
3.	Laufen auf 6 Beinen	S. 3
3.1	Gangart der Insekten	S. 3
3.2	Modell Mike	S. 3
3.2.1	Konstruktion des Modells	S. 3
3.2.2	Das erste Programm	S. 3
3.2.3	Die Linksdrehung	S. 4
3.2.4	Links, rechts, vor, zurück	S. 4
3.2.5	Hindernisse erkennen	S. 5
3.3	Modell Jack	S. 6
3.3.1	Die Konstruktion	S. 6
3.3.2	Die Programmierung	S. 6
4.	Laufen auf 4 Beinen	S. 8
4.1	Gangarten der Säugetiere	S. 8
4.2	Modell Joe	S. 8
4.2.1	Die Konstruktion	S. 8
4.2.2	Die Programmierung	S. 8
5.	Laufen auf 2 Beinen	S. 9
5.1	Zweibeinige Läufer	S. 9
5.2	Modell Jim	S. 9
6.	Zusammenfassung	S. 10

GB+USA CONTENTS

1.	Bionic – Nature as a Model	S. 12
2.	Requirements and Startup	S. 12
3.	Walking on Six Legs	S. 13
3.1	The Way Insects Walk	S. 13
3.2	Model Mike	S. 13
3.2.1	Assembly of the Model	S. 13
3.2.2	The First Program	S. 13
3.2.3	Turning Left	S. 14
3.2.4	Left, Right, Forward, Backward	S. 14
3.2.5	Detecting Obstacles	S. 15
3.3	Model Jack	S. 16
3.3.1	The Design	S. 16
3.3.2	The Programming	S. 16
4.	Walking on Four Legs	S. 18
4.1	The Way Mammals Walk	S. 18
4.2	Model Joe	S. 18
4.2.1	The Design	S. 18
4.2.2	The Programming	S. 18
5.	Walking on Two Legs	S. 19
5.1	Two-Legged Walkers	S. 19
5.2	Model Jim	S. 19
6.	Summary	S. 20

F SOMMAIRE

1.	Bionique – la Nature comme modèle	S. 22
2.	Conditions requises et mise en route	S. 22
3.	Marche sur 6 pattes	S. 23
3.1	Mode de déplacement des insectes	S. 23
3.2	Maquette Mike	S. 23
3.2.1	Conception de la maquette	S. 23
3.2.2	Le premier programme	S. 23
3.2.3	Rotation vers la gauche	S. 24
3.2.4	Gauche, droite, en avant, en arrière	S. 24
3.2.5	Reconnaître les obstacles	S. 25
3.3	Maquette Jack	S. 26
3.3.1	La conception	S. 26
3.3.2	La programmation	S. 26
4.	Marche sur 4 pattes	S. 28
4.1	Modes de déplacement des mammifères	S. 28
4.2	Maquette Joe	S. 28
4.2.1	La conception	S. 28
4.2.2	La programmation	S. 28
5.	Marche sur 2 pieds	S. 29
5.1	Les bipèdes	S. 29
5.2	Maquette Jim	S. 29
6.	Résumé	S. 30

NL INHOUD

1.	Bionic – de natuur als voorbeeld	S. 32
2.	Voorwaarden en voorbereiding	S. 32
3.	Lopen op zes benen	S. 33
3.1	Gang van de insecten	S. 33
3.2	Model Mike	S. 33
3.2.1	De constructie	S. 33
3.2.2	Het model programmeren	S. 33
3.2.3	Linksom draaien	S. 34
3.2.4	Links, rechts, vooruit, achteruit	S. 34
3.2.5	Hindernissen herkennen	S. 35
3.3	Model Jack	S. 36
3.3.1	De constructie	S. 36
3.3.2	Het model programmeren	S. 36
4.	Lopen op vier benen	S. 38
4.1	Gangen van de zoogdieren	S. 38
4.2	Model Joe	S. 38
4.2.1	De constructie	S. 38
4.2.2	Het model programmeren	S. 38
5.	Lopen op twee benen	S. 39
5.1	Tweebeinige lopers	S. 39
5.2	Model Jim	S. 39
6.	Samenvatting	S. 40

E CONTENIDO

1.	Biónica – La naturaleza como modelo	S. 42
2.	Requisitos e iniciación	S. 42
3.	Andar con 6 patas	S. 43
3.1	Modo de andar de los insectos	S. 43
3.2	Modelo Mike	S. 43
3.2.1	Construcción del modelo	S. 43
3.2.2	Primer programa	S. 43
3.2.3	Giro a la izquierda	S. 44
3.2.4	Izquierda, derecha, adelante, hacia atrás	S. 44
3.2.5	Reconocer obstáculos	S. 45
3.3	Modelo Jack	S. 46
3.3.1	Construcción	S. 46
3.3.2	Programación	S. 46
4.	Andar con 4 patas	S. 48
4.1	Modos de andar de los mamíferos	S. 48
4.2	Modelo Joe	S. 48
4.2.1	Construcción	S. 48
4.2.2	Programación	S. 48
5.	Andar con 2 patas	S. 49
5.1	Andadores de 2 patas	S. 49
5.2	Modelo Jim	S. 49
6.	Resumen	S. 50

P CONTÉUDO

1.	Bionic – A Natureza como modelo	S. 52
2.	Requisitos e primeiros passos	S. 52
3.	Andar sobre 6 pernas	S. 53
3.1	O andar dos insetos	S. 53
3.2	Modelo Mike	S. 53
3.2.1	Construção do modelo	S. 53
3.2.2	O primeiro programa	S. 53
3.2.3	A rotação para a esquerda	S. 54
3.2.4	Esquerda, direita, em frente, à ré	S. 54
3.2.5	Reconhecer obstáculos	S. 55
3.3	Modelo Jack	S. 56
3.3.1	A construção	S. 56
3.3.2	A programação	S. 56
4.	Andar sobre 4 pernas	S. 58
4.1	Andaduras dos mamíferos	S. 58
4.2	Modelo Joe	S. 58
4.2.1	A construção	S. 58
4.2.2	A programação	S. 58
5.	Andar sobre duas pernas	S. 59
5.1	Andante bípede	S. 59
5.2	Modelo Jim	S. 59
6.	Resumo	S. 60

1. Bionic – de natuur als voorbeeld

Het begrip „bionic“ bestaat uit de twee begrippen biologie en techniek. Deze tak van de wetenschap probeert zich voortdurend aan de natuur te oriënteren om technische problemen op te lossen.

Om zich steeds verder, sneller en effectiever voort te bewegen dan de natuur toestond, heeft de mens steeds weer machines uitgevonden die hier overeenkomstig de gestelde eisen op verschillende manieren voor zorgen. Voertuigen rijden op wielen. Op moeilijk terrein, waarop voertuigen op wielen zich niet kunnen verplaatsen, worden rupsvoertuigen ingezet. Schepen zwemmen op het water of kunnen duiken. Voor een aantal manieren van voortbeweging dient ook de natuur als voorbeeld. Zo lijkt een vliegtuig op een zwevende vogel.

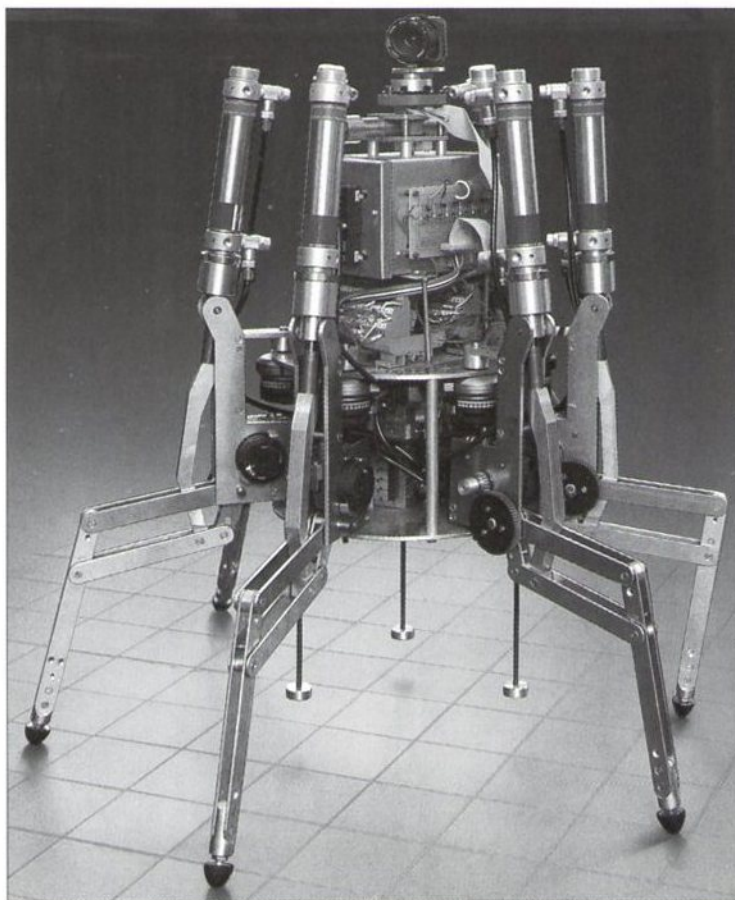
Sinds een aantal jaren houden wetenschappers zich met een andere manier van voortbeweging bezig die in de natuur veelvuldig voorkomt, namelijk het lopen. Ze ontwikkelden robots die zich op benen kunnen voortbewegen. Deze loomachines zouden overal kunnen worden ingezet waar voertuigen op wielen of rupsvoertuigen nauwelijks van nut zijn, bijv. op extreem oneffen of zacht terrein, om over hindernissen te klimmen, trappen op te lopen, sloten te passeren of om te worden ingezet op moeilijk toegankelijke en gevaarlijke plaatsen in kerncentrales, mijnen of tijdens reddingsacties.

De eerste serieuze pogingen om loomachines te ontwikkelen werden in 1967 ondernomen aan een universiteit in Tokio. Voor het eerst oriënteerden de wetenschappers zich aan de gang van de mens in plaats van aan insecten. De voortdurende verdere ontwikkeling van deze experimenten leverde in 1985 de eerste tweebeinige loomachine op. Inmiddels beschikken deze robots over ruim 50 vrijheidsgraden en een groot aantal microprocessors. Met behulp van een camera kunnen ze bijv. noten lezen en orgel spelen. Je kunt zelfs een gesprek met ze voeren.

Een voorbeeld van een zesbenige looprobot is "Achille", een elektro-pneumatische looprobot die aan de Koninklijke Militaire Academie in Brussel is ontwikkeld.

Uitgerust met een camera boven en aan de zes benen moet deze robot mechanisch op hoger en lager gelegen hindernissen (voorwerpen of kuilen) kunnen reageren.

Nu heeft fischertechnik zich ook met dit spannende onderwerp bezig gehouden en lopende robots geconstrueerd die vervolgens met de Intelligent Interface en de software LLWin „tot leven“ worden gewekt.



2. Voorwaarden en voorbereiding

Om de modellen van het computing-bouwpakket „Bionic Robots“ te kunnen bouwen, heb je behalve het bouwpakket nog de onderstaande artikelen nodig:

Intelligent Interface, art.-nr. 30402

Software LLWin (vanaf versie 3.0), art.-nr. 30407

Accu Set, art.-nr. 34969

Als je nog nooit met de software LLWin en de interface hebt gewerkt, kun je beter eerst het handboek van de software LLWin doorlezen. Daarin staat beschreven hoe je de software installeert en de interface aansluit. Bovendien is dit een prima manier om alvast een beetje ervaring op te doen hoe je modellen van fischertechnik via de PC bestuurt. Met een paar bouwcomponenten uit het bouwpakket (motor en knoppen) kun je om te beginnen eenvoudige modelbesturingen opbouwen.

Zodra je vertrouwd bent met de software en de interface, kun je je beslist aan de moeilijkere modellen Bionic Robots wagen.

In het bouwpakket zit tevens een CD-ROM met LLWin-voorbeeldprogramma's voor de modellen van het bouwpakket. Om deze programma's te kunnen openen, heb je de software LLWin vanaf versie 3.0 nodig. Je kunt de voorbeeldprogramma's op de CD laten en vanuit LLWin met het commando PROJET OPEN oproepen of de complete map BIONIC_ROBOTS van de CD in de projectdirectory van LLWin op de harde schijf kopiëren en de

voorbeelden vanaf deze locatie openen.

Voordat je de modellen bouwt, moet je ook nog een aantal onderdelen monteren, zoals het snoer en de stekker. Hoe je dat precies moet doen staat in de bouwhandleiding beschreven.

Zo, nu is het zover.

Welkom in de fascinerende wereld van de lopende fischertechnik-robots. Zodra je het eerste model af hebt en het bijna als een spook begint te bewegen, zul je verstand staan van deze techniek die in de natuur al miljoenen jaren wordt toegepast voor de voortbeweging.

3. Lopen op zes benen

3.1 Gang van de insecten

De gang van insecten is uitermate geschikt als voorbeeld voor de aandrijving van „mechanische zesbeners“. Bij de zgn. drievoetsgang worden steeds drie van de zes benen tegelijkertijd van de grond getild, het voorste en achterste been van de ene kant samen met het middelste been van de andere kant:

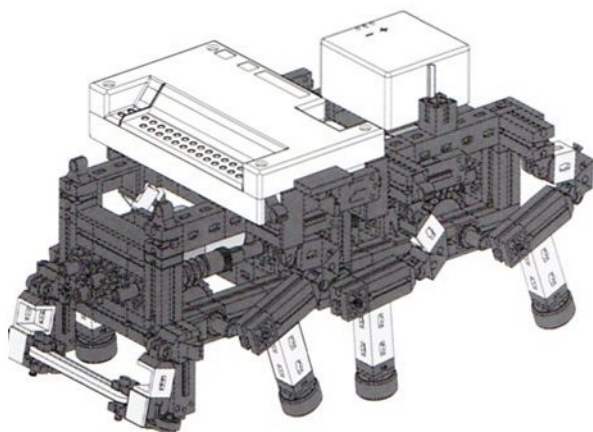


De benen die op de grond staan (zwarte blokjes) vormen een stabiele driepoot, zodat het model steeds stevig staat en niet omvalt tijdens het lopen.

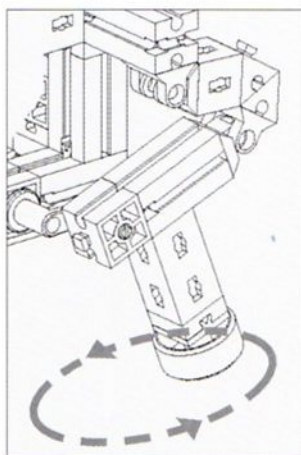
3.2 Model Mike

3.2.1 De constructie

Zet nu het zesbenige model Mike (zie bouwhandleiding p. 4) in elkaar. Laad tijdens het bouwen het accupack op, zodat je later voldoende energie hebt voor de aandrijving van het model.



De benen van het model zijn zodanig geconstrueerd dat ze een overbrenging via vier gewrichten vormen. De bouwvorm van het hier gebruikte viervoudige gewricht wordt ook wel kruk- en zwenkdrijfwerk genoemd. De beweegbaar gelagerde delen van de aandrijving voeren, aangedreven door een kruk, slingerende bewegingen uit. De afstanden tussen de afzonderlijke gewrichten en de stand van het voetpunt (het onderste deel van het been) zijn zo gekozen dat het voetpunt een elliptische beweging maakt als de aandrijfkruk draait. Hierdoor ontstaat een beweging die lijkt op een stap tijdens het lopen. De zes krukken die de benen aandrijven moeten precies volgens de bouwhandleiding worden afgesteld. De drie benen die



tegelijkertijd op de grond neerkomen, hebben dezelfde krukstand. De krukken van de drie benen, die op dit moment in de lucht staan, zijn daarvoor 180° gedraaid. De juiste stand van de krukken ten opzichte van elkaar zorgt ervoor dat het model op de juiste manier, namelijk in de drievoetsgang, kan lopen.

De moeren waarmee de wormen en tandwielen op de assen worden gefixeerd, moeten goed worden aangehaald, zodat de krukken tijdens het lopen op hun plaats blijven.

De rechter- en linkerkant van het model worden elk door een afzonderlijke motor aangedreven (dit is noodzakelijk om bochten te kunnen maken). Om die reden is het van belang dat het middelste been van de ene kant zich altijd in dezelfde positie bevindt als de twee buitenste benen aan de andere kant. Deze synchronisatie wordt gestuurd door de software via de knoppen E1 en E2.

Test met behulp van de interfacediagnose (Check Interface) of alle knoppen en motoren juist zijn aangesloten. Draairichting van de motoren: linkse draairichting = vooruit.

3.2.2 Het model programmeren

Nu gaan we Mike iets aanleren. Om te beginnen is het voldoende als het model vooruit loopt. Bochten maken en reageren op hindernissen komt later aan de orde.

Opgave 1:

Programmeer het model zodanig dat het in de drievoetsgang recht vooruit loopt. Gebruik de knoppen E1 en E2 om de benen aan de linker- en rechterkant te synchroniseren. Let erop dat de twee buitenste benen van de ene kant altijd in dezelfde stand staan als het middelste been van de andere kant. Gebruik bovendien knop E8 als resetknop.

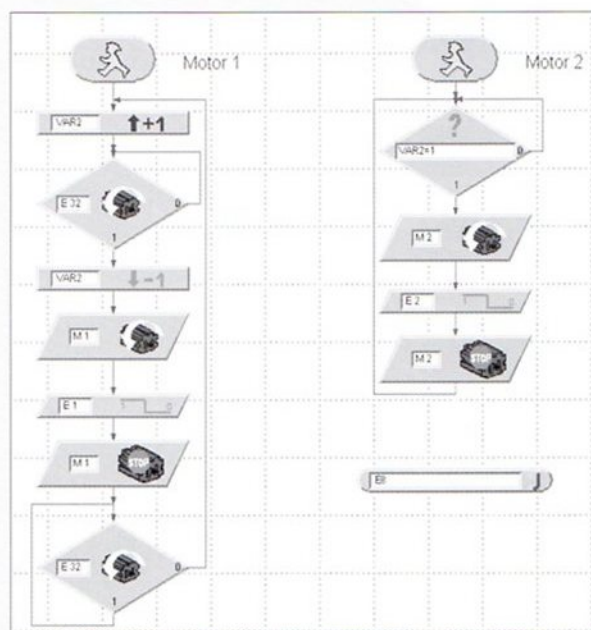
Tips:

Programmeer voor iedere motor een eigen afloop. Stuur de afloop van motor M2 met behulp van een variabele VAR2.

Als je tijdens het programmeren geen gebruik maakt van de interface, kun je de stroomverbinding tussen het accupack en de interface onderbreken om energie te sparen.

Oplossing:

Het programma voor recht vooruitlopen ziet er als volgt uit:



De variabele VAR2 geeft de impuls dat motor M2 start. Daarna wordt motor M1 gestart. Zodra schakelaar E1 wordt ingedrukt, stopt M1. Zodra E2 wordt ingedrukt, stopt M2. De eerste afloop wacht tot M2 is gestopt (toestand van de motor M2 wordt via E32 opgevraagd; zie ook „Checking Motor Conditions” in het handboek LLWin).

Wie trouwens niet zoveel moeite wil doen om deze afloop zelf te maken, vindt hem als voorbeeldproject MIKE_VOORUIT.MDL op de bijgevoegde CD. Start het project op. Als je alles goed hebt geprogrammeerd, komt er beweging in het model en loopt het vooruit. Van harte gefeliciteerd! De eerste stap is gezet.

3.2.3 Linksom draaien

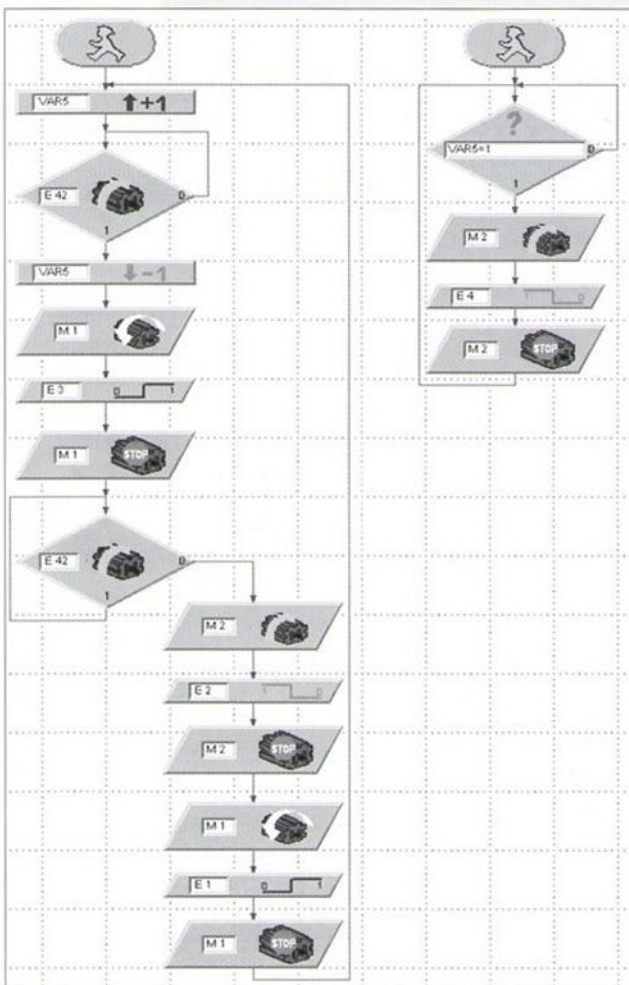
We zijn natuurlijk nog niet tevreden met het feit dat Mike alleen maar vooruit loopt. Nu willen we dat hij zich op de plaats omdraait.

Opgave 2:

Programmeer Mike zodanig dat hij linksom draait.

Tips:

Het model draait linksom als M1 linksom en M2 rechtsom draait. Je kunt het model natuurlijk gebruiken zonder dat het is gesynchroniseerd. Het kan dan weliswaar omdraaien, maar er zijn bepaalde posities waarin het model voorover valt. Dat kun je voorkomen. En wel met de volgende stappen:



Met behulp van de schakelaars E1-E4 bewegen de linker- en rechterkant van het model eerst één stap tegelijk, vervolgens zet de linkerkant één stap, daarna de rechterkant etc. Op die manier valt het model nooit voorover. Probeer het eens! Zo is ook de volgorde gemakkelijker te begrijpen. Ook deze afloop kun je als project MIKE_LINKS.MDL op de CD vinden.

Nu kan het model vooruit lopen en linksom draaien. Wat er nog ontbreekt is achteruit lopen en rechtsom draaien. Het achteruit lopen werkt in principe als het vooruit lopen, maar dan met de omgekeerde draairichting van de motor. Het rechtsom draaien is in feite het omgekeerde principe van het linksom draaien.

3.2.4 Links, rechts, vooruit, achteruit

Opgave 3:

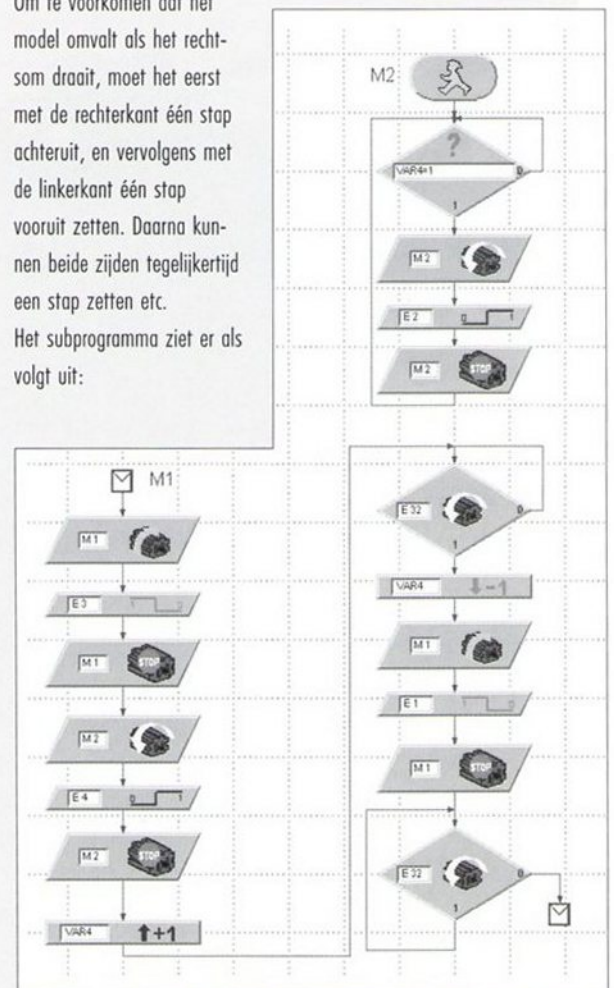
Programmeer nu de functies VOORUIT, ACHTERUIT, LINKS en RECHTS als subprogramma, zodat je ze later in verschillende projecten flexibel kunt inzetten.

Tips:

Hoe je een bestaande afloop in een subprogramma kopieert, staat beschreven in het LLWin-handboek.

Gebruik in ieder subprogramma een andere variabele (VAR2-VAR5) om de afloop voor motor M2 te starten.

Om te voorkomen dat het model omvalt als het rechtsom draait, moet het eerst met de rechterkant één stap achteruit, en vervolgens met de linkerkant één stap vooruit zetten. Daarna kunnen beide zijden tegelijkertijd een stap zetten etc. Het subprogramma ziet er als volgt uit:



De andere subprogramma's hebben we hier niet afgedrukt. Als het programmeren van een afloop moeilijkheden oplevert, vind je de subprogramma's kant en klaar in het bestand MIKE_VOORBEELD.MDL op de CD. Het hoofdprogramma van dit project is leeg. In het "Toolbox" onder de indexkaart "Subprograms" vind je een lijst met de beschikbare subprogramma's die je in het hoofdprogramma kunt invoegen.

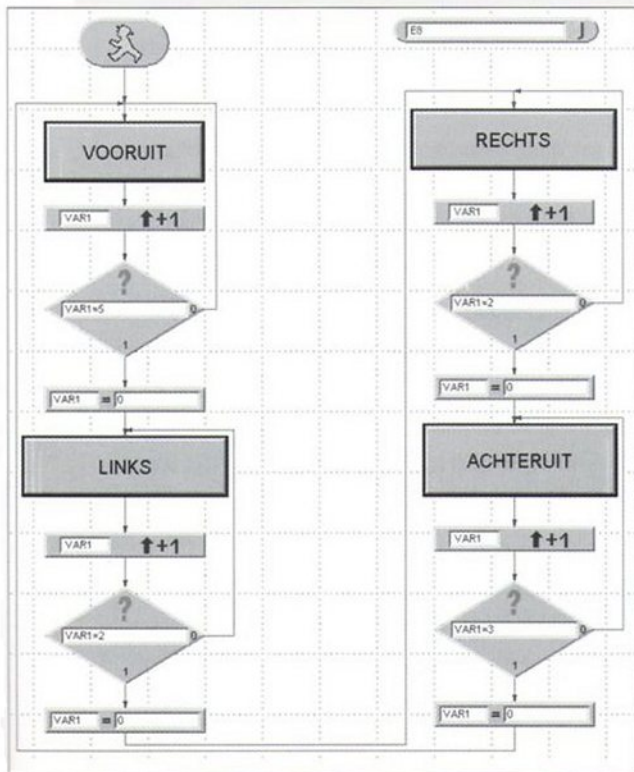


Maar kijk niet meteen na hoe het moet. Probeer eerst zelf achter de oplossing te komen. Als het niet lukt, kun je het immers nog altijd opzoeken. Om alle subprogramma's eens te testen willen we Mike nu laten dansen.

Opgave 4:

Programmeer Mike zodanig dat hij 5 stappen naar voren zet, twee stappen naar links draait, dan twee stappen naar rechts, vervolgens 3 stappen achteruit zet en daarna opnieuw begint. Gebruik als telvariabele voor het aantal stappen de variabele Var1. Gebruik E8 als resetknop.

Oplossing:



Dit project heet MIKE_DANS.MDL.

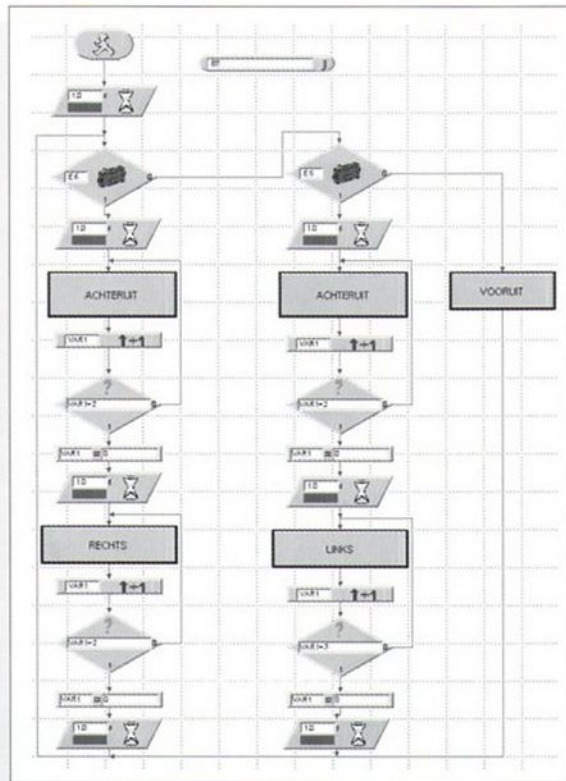
3.2.5 Hindernissen herkennen

Ten slotte willen we Mike nog zover krijgen dat hij met zijn beweegbare stoterstang (of liever „voeler“) hindernissen herkent en uitwijkt.

Opgave 5a:

Programmeer Mike zodanig dat hij bij een hindernis aan zijn linkervoeler (knop E6) eerst 4 stappen achteruit en dan 2 stappen naar rechts uitwijkt. Als er sprake is van een hindernis aan zijn rechervoeler (knop E5) moet hij 4 stappen achteruit en vervolgens 3 stappen naar links uitwijken.

Oplossing:



Mike loopt eerst altijd rechtdoor. Na iedere stap worden de knoppen E5 en E6 gepolst. Als E6 is ingedrukt, springt het programma naar de linkerafloop (eerst achteruit, dan naar rechts). Als E5 is ingedrukt, springt het naar de middelste afloop (eerst achteruit, dan naar links).

Aangezien de knoppen E5 en E6 alleen na iedere volledige stap worden gepolst, duurt het nogal lang totdat Mike op een hindernis reageert.

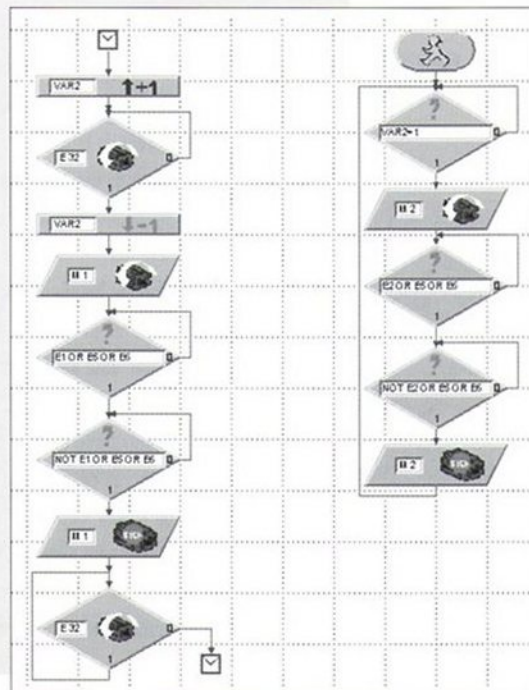
Opgave 5b:

Optimaliseer het subprogramma VOORUIT zodanig dat Mike sneller op een hindernis kan reageren.

Tip:

Gebruik voor het polsen van de knoppen E1 en E2 niet de component EDGE maar de component COMPARE. Pols daarmee bovendien of E5 of E6 zijn ingedrukt.

Oplossing:



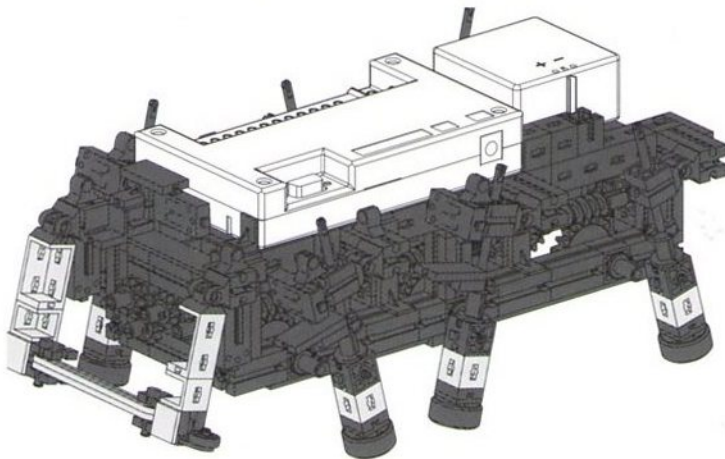
Nu zou Mike perfect moeten functioneren. Dit programma staat eveneens op de CD onder MIKE_HINDERNIS.MDL. Je kunt het verbeterde subprogramma ook in het project MIKE_VOORBEELD.MDL integreren. Als E5 en E6 in een ander programma niet worden gepolst, is dat helemaal niet erg. Dit verbeterde voorbeeld hebben we onder MIKE_VOORBEELD_HINDERNIS.MDL opgeslagen.

Nadat we nu de eerste zesbenige robot uitvoerig hebben behandeld, richten we onze aandacht op het tweede model dat eveneens zes benen heeft. We noemen het "Jack".

3.3 Model Jack

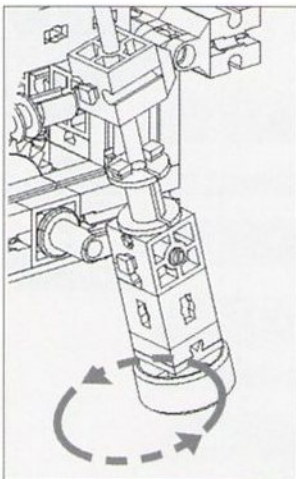
Jack maakt eveneens deel uit van het geslacht van de zesbenige fischertechnik-modellen. Hij heeft echter een volstrekt andere beenconstructie dan Mike.

Zet nu het model in elkaar volgens de bouwhandleiding vanaf p. 12. De stappen 1-13 zijn bij Mike en Jack overigens identiek. Je hoeft Mike dus niet volledig te demonteren voordat je begint Jack te bouwen.



3.3.1 De constructie

De beenconstructie van Jack is eveneens een overbrenging met vier gewrichten. De hier toegepaste bouwvorm wordt "slingerend kruk- sleufmechanisme" genoemd. De drijfstaaf is in een beweegbare langgeleider gelagerd die heen en weer slingert als de kruk draait. De cirkelbeweging die het voetpunt van het been maakt, is niet elliptisch als bij het model Mike, maar rond.



Hierdoor beweegt Jack's lichaam tijdens het lopen meer op en neer dan Mike. De stappen zijn ook korter. Jack kan echter in tegenstelling tot Mike hindernissen passeren. Bovendien doet deze overbrengingsbouwvorm meer aan een been denken dan bij Mike het geval is. Als Jack loopt, lijkt het alsof hij op stelten loopt.

Hij beweegt eveneens in de driehoetsgang net als insecten. Ook bij dit model is het belangrijk om de krukken precies volgens de bouwhandleiding af te stellen en de moeren goed aan te halen.

3.3.2 Het model programmeren

Het ligt voor de hand te denken dat voor Jack dezelfde programma's kunnen worden gebruikt als voor Mike. Probeer het maar eens!

Opgave 1:

Laat Jack met het programma MIKE_HINDERNIS.MDL lopen. Wat gebeurt er?

Waarneming:

Het model loopt foutloos vooruit en achteruit. Maar als het naar links en rechts draait, valt het voorover.

Opgave 2:

Hoe verklaar je dat?

Oplossing:

De twee modellen beschikken over verschillende beenconstructies. Bovendien worden de knoppen E1-E4 op een andere positie van het been ingedrukt. De manier waarop Mike draait, hoeft derhalve nog lang niet geschikt te zijn voor Jack – helaas.

Daar nemen we natuurlijk geen genoegen mee en we gaan zo snel mogelijk een oplossing vinden.

Opgave 3:

Probeer Jack zodanig te programmeren dat hij net als Mike hindernissen herkent, maar niet voorover valt als hij draait.

Tips:

Sla het project MIKE_HINDERNIS.MDL op als JACK_HINDERNIS.MDL en breng de noodzakelijke wijzigingen aan.

Als Jack draait, kunnen de twee motoren steeds tegelijkertijd lopen. Het afwisselend in- en uitschakelen vervalt. Het kritieke punt is dat de benen zich aan het begin van de draaibeweging, dus na het achteruit lopen, in de goede uitgangspositie bevinden.

Linksom draaien:

Als het model linksom moet draaien, moet de kruk van het voorste linkerbeen aan het begin van de draaibeweging naar achter en de kruk van het voorste rechterbeen naar voren wijzen. Dit is het geval als tijdens het achteruit lopen aan de linkerkant de knop E2 en aan de rechterkant de knop E1 wordt ingedrukt en weer los wordt gelaten.

Rechtsom draaien:

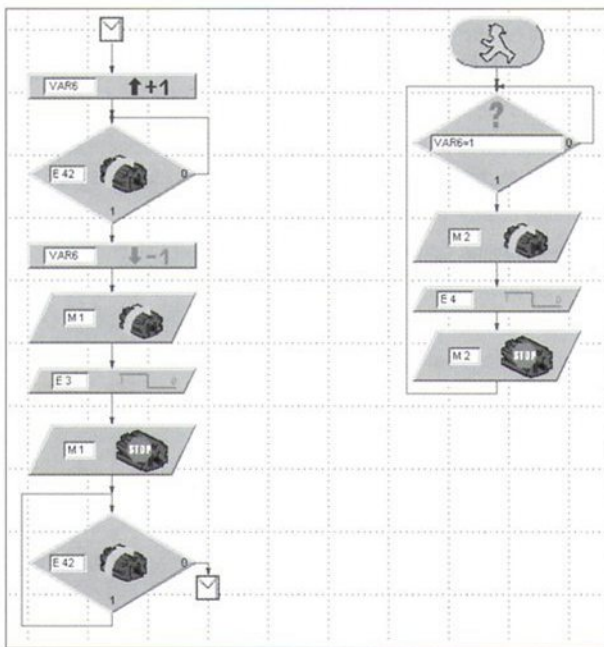
Als het model rechtsom moet draaien, moet de kruk van het voorste linkerbeen aan het begin van de draaibeweging naar voren en de kruk van het voorste rechterbeen naar achter wijzen. Dit is het geval als tijdens het achteruit lopen aan de linkerkant de knop E4 en aan de rechterkant de knop E3 wordt ingedrukt en weer los wordt gelaten.

Tamelijk ingewikkeld, hè? Maar we zijn er bijna:

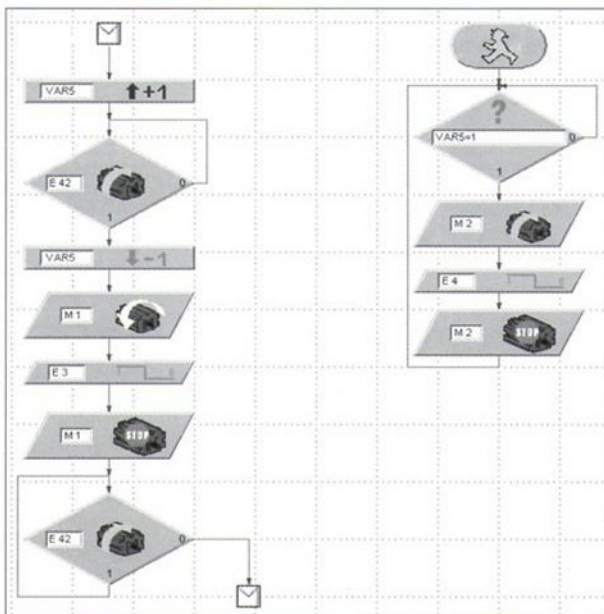
Voor de beweging achteruit moeten twee verschillende subprogramma's worden gebruikt.

Als het model naar links moet draaien, synchroniseer je de stappen tijdens het achteruit lopen met behulp van de knoppen E1 en E2. Dit komt overeen met het subprogramma ACHTERUIT uit het project MIKE_HINDERNIS.MDL. Als het model naar rechts moet draaien, worden de stappen achteruit met behulp van E3 en E4 gesynchroniseerd.

Je slaat dus het programma ACHTERUIT met het commando SUBPROGRAM - RENAME onder de naam ACHTERUIT_L op. Vervolgens kopieer je het met SUBPROGRAM - COPY in een tweede subprogramma ACHTERUIT_R. Daar verander je de knopdefinities voor het synchroniseren in E3 en E4. Vergeet niet voor ACHTERUIT_R een nieuwe variabele VAR6 voor het synchroniseren te gebruiken, anders gaat het behoorlijk mis. ACHTERUIT_R ziet er dan als volgt uit:

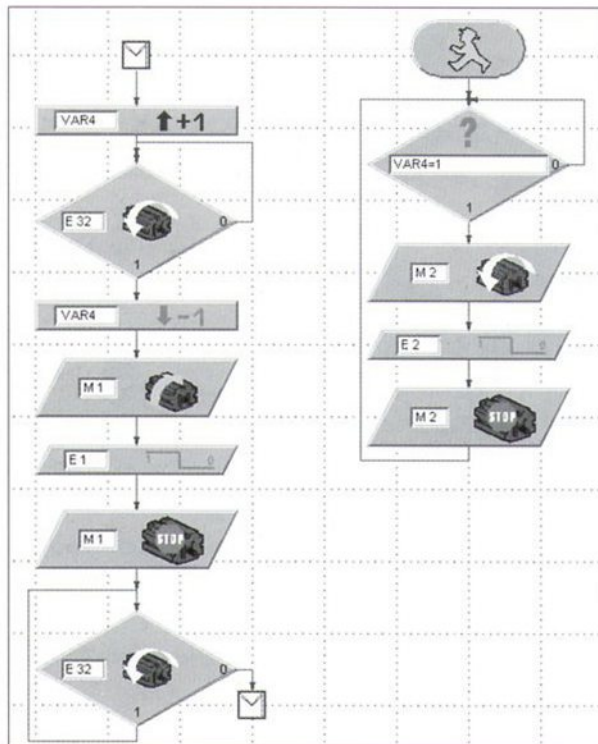


Nu moeten nog de subprogramma's voor het draaien zelf worden gewijzigd, zodat beide motoren steeds tegelijkertijd draaien. Het subprogramma LINKS bestaat uit de volgende componenten:

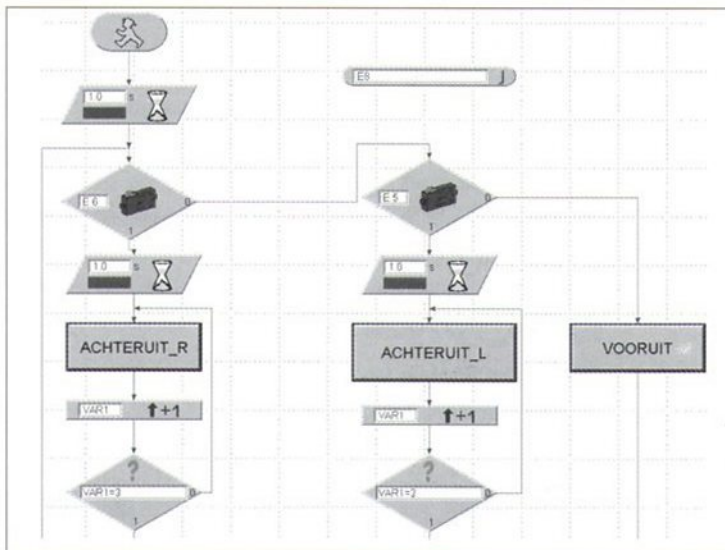


Je ziet dat ten opzichte van het subprogramma in het project MIKE_HINDERNIS.MDL een aantal componenten kunnen vervallen.

Het subprogramma voor rechtsom draaien ziet er ongeveer hetzelfde uit, alleen met andere draairichtingen van de motoren. Bovendien worden de knoppen E1 en E2 gebruikt om de motoren te synchroniseren:



Als laatste vervang je in het hoofdprogramma in de onderverdeling voor het uitwijken naar rechts het subprogramma ACHTERUIT_L door ACHTERUIT_R:



Het hoofdprogramma blijft voor de rest ongewijzigd.

Klaar! Als je nergens een fout hebt gemaakt, kan Jack nu lopen zonder om te vallen bij het draaien. Als iets niet functioneert en je er niet achterkomt waarom, moet je daar niet te zwaar aan tillen, het was immers bepaald geen gemakkelijke klus. Je hebt in ieder geval nog de mogelijkheid om het uitgewerkte project JACK_HINDERNIS.MDL gewoon van de CD op te roepen en het model daarmee te besturen.

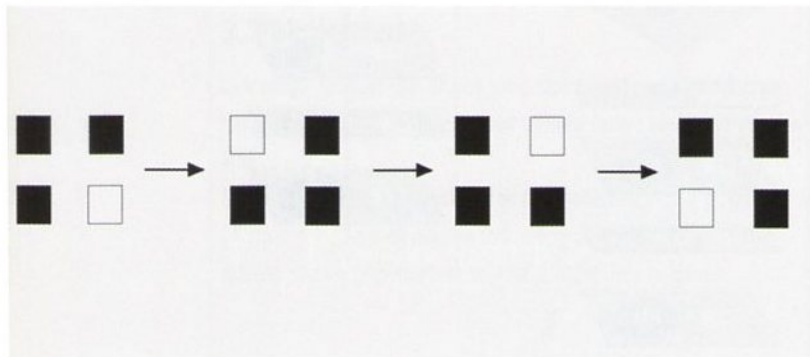
Als je het probleem zelf kon oplossen, mag je trots zijn, want dan ben je vanaf nu een professionele programmeur.

4. Lopen op vier benen

4.1 Gangen van de zoogdieren

Om een looprobot met vier benen te construeren nemen we de natuur weer als voorbeeld en kijken we eens hoe zoogdieren zich voortbewegen.

De langzaamste en veiligste gang is de zogeheten pas. Eén been zoekt een nieuwe plaats om op te staan, terwijl het lichaam van het dier op drie benen steunt. De dieren verplaatsen hun benen kruiselings in de volgorde: rechtervoorbeen, linkerachterbeen, linkervoorbeen, rechterachterbeen vooruit.

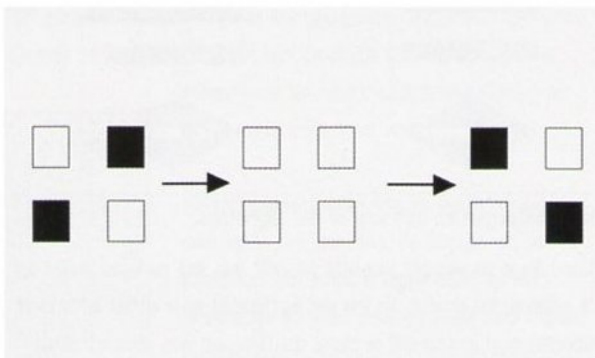


De zwarte vlakken zijn de benen die op de grond staan, de witte vlakken het opgetilde been.

Als we deze gang bij een looprobot willen toepassen, moeten we over het volgende nadenken:

Stel je voor dat je een been afzaagt van een tafel met vier poten. Wat gebeurt er? Juist, de tafel valt om. De drie benen vormen dus geen stabiele driepoot meer zoals bij de zesbenige soortgenoten het geval was. Dat maakt het construeren van een vierbenige robot moeilijker.

Hoe sneller de zoogdieren zich voortbewegen, hoe onstabiel hun gang wordt. Laten we nog even naar de gang „draf” kijken. Hierbij worden de benen diagonaal synchronoos opgetild. Voordat ze echter de grond raken, worden de twee andere benen al opgetild. Dit betekent dat tijdelijk het bodemcontact volledig verloren gaat.

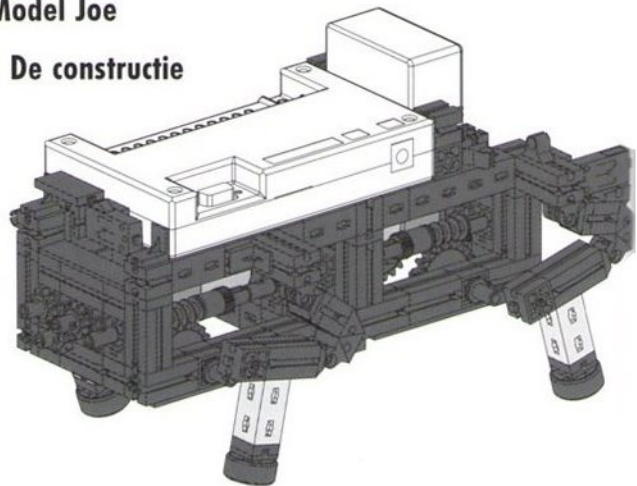


Je kunt je zeker voorstellen dat een gang waarbij het bodemcontact tijdelijk verloren gaat voor een fischertechnik-model, waarop de interface en accupack zijn aangebracht, niet bepaald geschikt is.

Laten we het dus maar eens proberen met de gang „stap”.

4.2 Model Joe

4.2.1 De constructie



Zet het model in elkaar zoals in de bouwhandleiding vanaf p. 20 staat uitgelegd.

De constructie van de benen is dezelfde als bij Mike. De stand van de krukken die de benen aandrijven moet bij Joe volledig anders zijn.

De krukken zijn ten opzichte van elkaar steeds 90° verdraaid. Je moet ze precies volgens de bouwhandleiding afstellen. Het synchroniseren van de linker- en rechterkant geschiedt wederom via de knoppen E1 en E2. Zodoende bereiken we de vereiste opeenvolging van stappen.

Om te voorkomen dat het model omvalt zodra een been wordt opgetild, moet het zwaartepunt van het model zo liggen dat het model op het juiste moment kantelt en door het zojuist ontlaste been wordt opgevangen.

4.2.2 Het model programmeren

Bij dit model nemen we genoeg met het vooruit lopen.

Opgave 1:

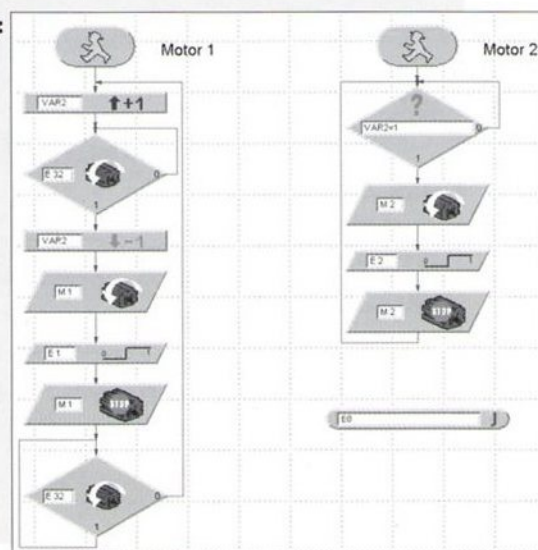
Programmeer Joe zodanig dat hij in de gang „stap” vooruit beweegt.

Tips:

Gebruik voor elke motor een afzonderlijke afloop en synchroniseer de motoren met behulp van de knoppen E1 en E2.

Gebruik E8 als resetknop.

Oplossing:



Bij dit programma gebruiken we de 0-1 zijkant van de knoppen voor het synchroniseren. Op het moment dat de knoppen zijn ingedrukt, hebben de krukken van de 4 benen de juiste positie ten opzichte van elkaar. Dit project noemen we JOE.MDL.

Je ziet dat Joe veel moeizamer beweegt dan Mike en Jack. Door de noodzakelijke gewichtsverplaatsing schommelt het lichaam nogal sterk en zijn gang is zeker niet zo elegant als die van onze zesbenige vrienden.

Misschien wil je dit model nog aanleren om bochten te nemen, probeer gewoon of het lukt. Veel succes.

5. Lopen op twee benen

5.1 Tweebenige lopers

Het lopen op twee benen is niet in het geslacht van de zoogdieren ontstaan, maar komt ook bij een aantal reptielsoorten voor. Varanen, leguanen, agamen enz. gebruiken tijdens de vlucht alleen hun achterpoten. Op die manier bereiken ze zeer hoge snelheden. Daartoe hebben ze stevige achterpoten nodig, een lange staart om in evenwicht te blijven en een vlak terrein.

Vogels zijn eveneens tweebeners. De struisvogel hoort tot de snelste loopvogels. Hij bereikt snelheden tot wel 60 km/h.

De meest perfecte tweebener is de mens. Om rechtop te kunnen lopen moet het heupgewricht gestrekt zijn. Hiervoor wordt gezorgd door de grote bilspier. Bovendien kunnen de benen in het kniegewricht worden "vastgezet" en zodoende in een energiearme houding worden gefixeerd.

Het voortbewegen op twee benen is de moeilijkste manier, want ze veronderstelt naast de hierboven beschreven anatomische voorwaarden een zeer goed ontwikkeld evenwichtsgevoel. Voor ons is het vanzelfsprekend om op twee benen te lopen. Maar als we erbij stilstaan dat bij het optillen van één been het hele lichaam op slechts één been steunt en op die manier in balans moet worden gehouden, merken we dat juist het evenwicht behouden voor moeilijkheden zorgt bij deze manier van voortbeweging. Zelfs een pasgeboren soortgenoot kan niet meteen op twee benen lopen. Hij kruipt eerst "op armen en benen" voordat hij gaat staan en "leert te lopen".

Aan de Waseda-universiteit in Tokio werden reeds tweebenige robots ontwikkeld die met behulp van een groot aantal gewrichten, uiteenlopende sensoren, camera's en efficiënte microprocessors bewegen en in evenwicht blijven door hun gewicht te verplaatsen.

Voor ons bouw pakket Bionic Robots zou dat echter te kostbaar en ingewikkeld zijn. We hebben gezien dat we al bijna onze grenzen bereiken als een fischertechnik-model op vier benen loopt.

5.2 Model Jim

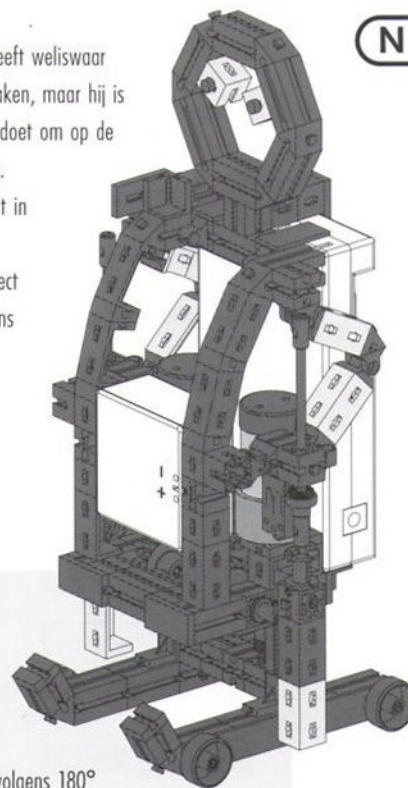
Om dit hoofdstuk echter niet alleen van de theoretische kant te bekijken, hebben we ten slotte besloten om ten minste een tweebenige skiër, we

noemen hem Jim, te construeren. Hij heeft weliswaar weinig met een tweebenige loper te maken, maar hij is een aardige kerel die zijn uiterste best doet om op de één of andere manier vooruit te komen.

Dit mag je niet missen. Het model staat in de bouwhandleiding op p. 27.

Als programma kun je gewoon het project JOE.MDL gebruiken. Je hoeft er niet eens iets aan te veranderen. Jim werkt ook met dit programma en kwakkel naar voren.

Eén opgave hebben we nog voor je:



Opgave 1:

Programmeer Jim zodanig dat hij ongeveer 50 cm vooruit loopt, vervolgens 180° rechtsom draait, hetzelfde stuk terug loopt (vooruit), dan weer 180° linksom draait, hetzelfde stuk terug loopt etc. Gebruik voor het aantal stappen vooruit de terminale parameter EA, voor het aantal stappen linksom EB en voor rechtsom EC. Gebruik weer E8 als resetknop.

Tips:

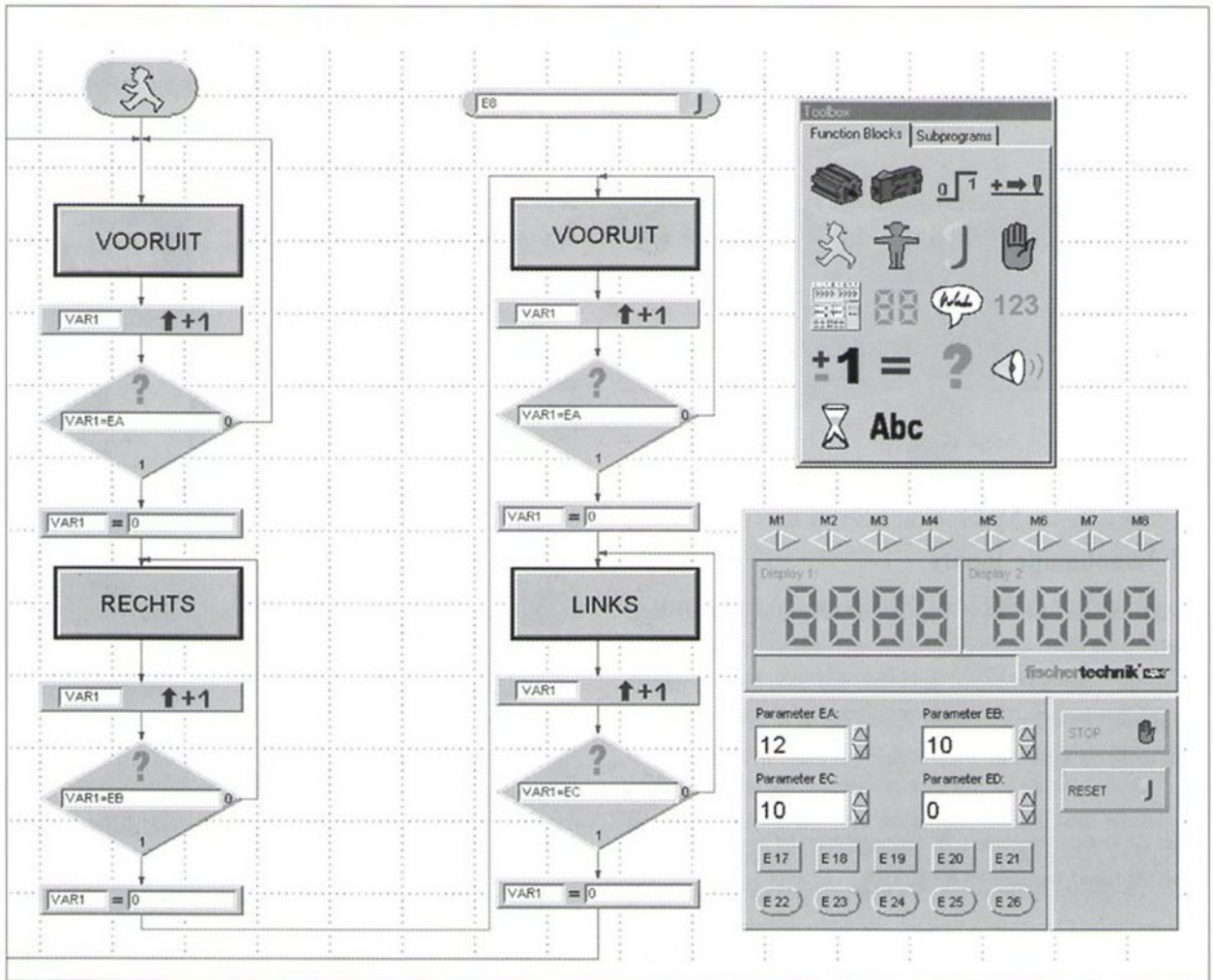
Sla het project JOE.MDL op als JIM.MDL. Maak daarin van het hoofdprogramma een subprogramma VOORUIT (componenten markeren en uitknippen, via EDIT – SUBPROGRAM een nieuw programma maken, componenten toevoegen, SUBIN en SUBOUT aanvullen, zie ook LLWin-handboek).

Maak van dit subprogramma de vereiste subprogramma's LINKS en RECHTS met het commando SUBPROGRAM – COPY. Wijzig hierin de draairichting van de motor en de opvraag van de draairichtingen van de motoren en gebruik voor ieder subprogramma een andere besturingsvariabele voor motor 2.

Vervolgens programmeer je het hoofdprogramma vergelijkbaar als bij MIKE_DANS.MDL. Het verschil is dat je nu de instelbare terminale parameters EA-EC gebruikt voor het aantal stappen. Hoeveel stappen Jim nodig heeft om 180° te draaien resp. een halve meter vooruit te komen, moet je even uitproberen.

Oplossing:

Hieronder volgt een schema van het hoofdprogramma. De subprogramma's kun je indien nodig direct op het beeldscherm bekijken. Bij ons heet het project eveneens JIM.MDL.



We hebben in het project ook meteen het subprogramma ACHTERUIT aangevuld, ook al is dat hier niet meteen nodig. Maar je wilt beslist dat Jim nog andere pasjes leert. Misschien moet hij daarbij ook een keer achteruit bewegen.

6. Samenvatting

Op je reis door de wereld van de Bionic Robots van fischertechnik ben je er zeker achter gekomen dat het niet altijd gemakkelijk was om onze vier vrienden te laten lopen. Het is nu eenmaal moeilijker om op benen te lopen dan op wielen te rollen. Met name het programmeren van de synchronisatie tussen de linker- en rechterkant bij het linksom of rechtsom draaien vereist een beetje concentratie. Maar voor iedereen die toch liever bezig is met het bouwen van de modellen, hebben we alle programma's kant en klaar op een CD gebrand, zodat iedereen de modellen kan bouwen en besturen.

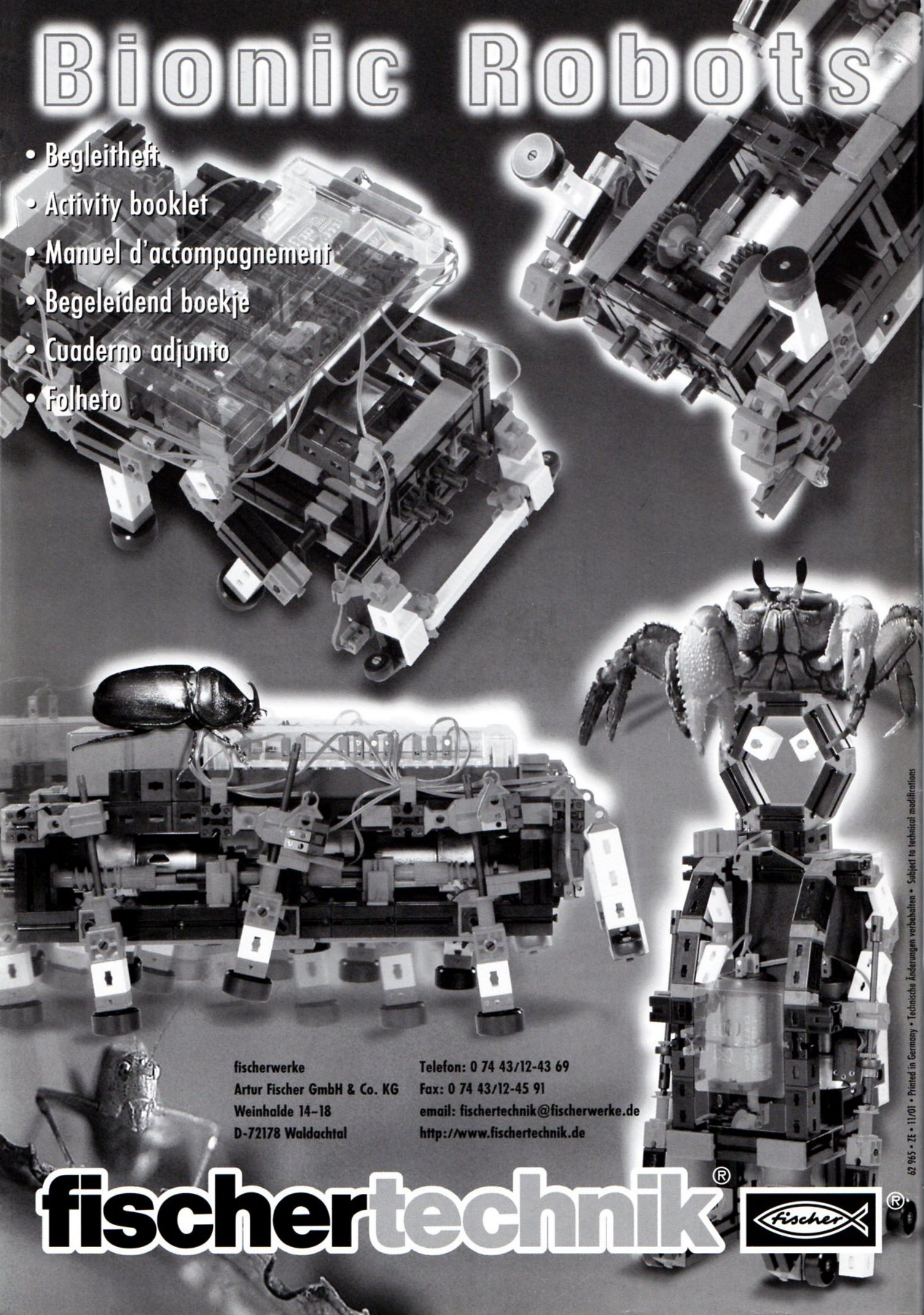
Als je jezelf tot de professionele programmeurs mag tellen, heb je beslist nog een heleboel ideeën om nog een aantal opgaven voor Mike, Jack, Joe of Jim te programmeren, hetzij met extra sensors, zodat ze niet van de tafel naar beneden vallen, of dat ze in een labyrint niet verdwalen.

Met behulp van extra componenten kun je de modellen ook nog voorzien van een hoofd, slurf of staart. Laat je fantasie de vrije loop. Maak er iets van!



Bionic Robots

- Begleitheft
- Activity booklet
- Manuel d'accompagnement
- Begeleidend boekje
- Cuaderno adjunto
- Folheto



fischerwerke
Artur Fischer GmbH & Co. KG
Weinhalde 14-18
D-72178 Waldachtal

Telefon: 0 74 43/12-43 69
Fax: 0 74 43/12-45 91
email: fischertechnik@fischerwerke.de
<http://www.fischertechnik.de>

fischertechnik®

