

# 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   | Zwei-beinige 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   | Twee-beinige 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 – Die Natur als Vorbild

Der Begriff Bionic setzt sich zusammen aus den beiden Begriffen Biologie und Technik. Dieser Zweig der Wissenschaft versucht stets, sich bei technischen Lösungen an der Natur zu orientieren.

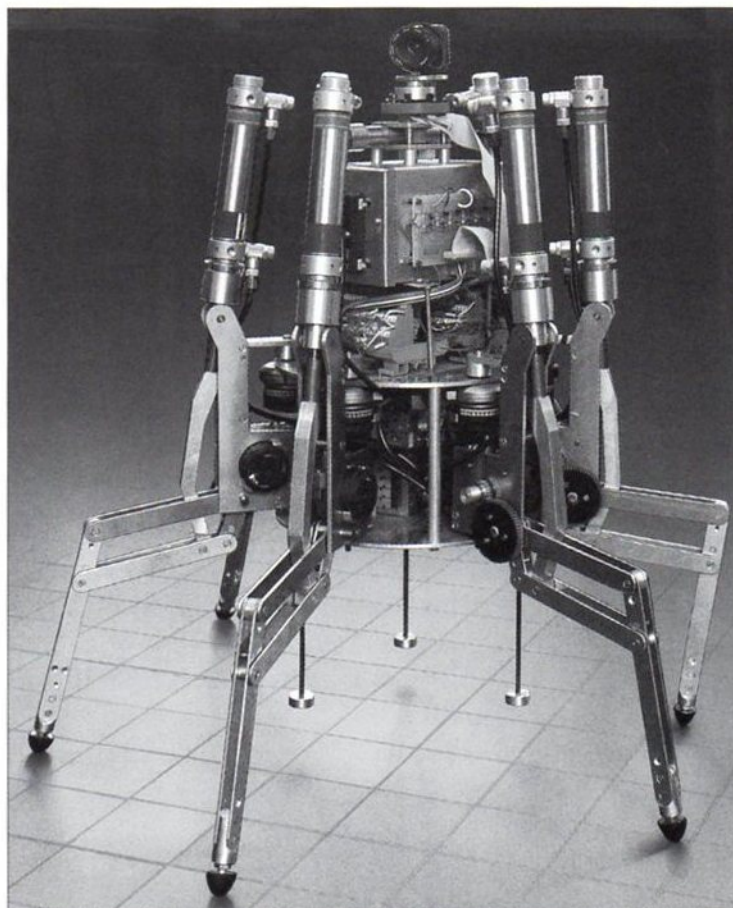
So hat der Mensch mit dem Ziel, sich weiter, schneller und effektiver fort zu bewegen als von Natur aus möglich, immer wieder Maschinen erfunden, die dies den jeweiligen Anforderungen entsprechend auf unterschiedliche Weise gewährleisten. Fahrzeuge rollen auf Rädern. In schwierigem Gelände, wo Radfahrzeuge versagen, werden Kettenfahrzeuge eingesetzt. Schiffe schwimmen auf dem Wasser oder sind in der Lage zu tauchen. Bei einigen Fortbewegungsarten dient auch die Natur als Vorbild. So gleicht beispielsweise ein Flugzeug einem segelnden Vogel.

Seit einigen Jahren beschäftigen sich Wissenschaftler mit einer weiteren, in der Natur sehr verbreiteten Bewegungsform, dem Gehen bzw. Laufen. Es werden Roboter entwickelt, die in der Lage sind, sich auf Beinen fort zu bewegen. Solche Laufmaschinen könnten überall dort eingesetzt werden, wo Rad- und Kettenfahrzeuge kaum mehr eine Chance hätten, so z. B. in äußerst unebenem oder nachgiebigem Gelände, beim Klettern über Hindernisse, Treppen steigen, Überwinden von Gräben oder beim Einsatz an schwer zugänglichen und gefährlichen Stellen in Kernkraftwerken, Bergwerkstollen oder bei Rettungsaktionen.

Die ersten ernsthaften Versuche bei der Entwicklung von Laufmaschinen entstanden 1967 an einer Universität in Tokio. Erstmals orientierte man sich an Stelle von Insekten an der menschlichen Gangart. Die ständige Weiterentwicklung dieser Versuche führte 1985 zu der ersten zweibeinigen Laufmaschine. Inzwischen besitzen diese Roboter über 50 Freiheitsgrade und zahlreiche Mikroprozessoren. Mit Hilfe einer Kamera können Sie z. B. Noten lesen und Orgel spielen. Man kann sich sogar mit ihnen unterhalten.

Ein Beispiel für einen sechsbeinigen Laufroboter ist der an der Königlichen Militärakademie in Brüssel entwickelte elektropneumatische Laufroboter „Achille“. Ausgestattet mit einer Kamera oben und an den sechs Beinen, soll dieser Roboter mechanisch auf erhöhte oder vertiefte Hindernisse (Gegenstände oder Löcher) reagieren.

Nun hat sich auch Fischertechnik diesem spannenden Thema gewidmet und laufende Roboter konstruiert, die dann mit dem Intelligent Interface und der Software LLWin „zum Leben erweckt“ werden.



## 2. Voraussetzungen und Einstieg

Damit du die Modelle des Computing-Baukasten „Bionic Robots“ bauen kannst, benötigst du zusätzlich zum Baukasten noch folgende Artikel:

**Intelligent Interface, Art.-Nr. 30402**

**Software LLWin (ab Version 3.0), Art.-Nr. 30407**

**Stromversorgung Accu Set, Art.-Nr. 34969**

Wenn du dich mit der Software LLWin und dem Interface noch nicht auskennst, solltest du zunächst das Handbuch der Software LLWin durchlesen. Dort ist beschrieben, wie die Software installiert und das Interface angeschlossen wird. Es ist außerdem bestens dazu geeignet, erste Erfahrungen zu sammeln, wie man Fischertechnik-Modelle über den PC steuert. Mit einigen wenigen Bauteilen aus dem Baukasten (Motor und Taster) kannst du dir zunächst ganz einfache Modellsteuerungen aufbauen.

Sobald du mit der Software und dem Interface vertraut bist, kannst du dich dann durchaus an die anspruchsvolleren Bionic-Robots-Modelle wagen.

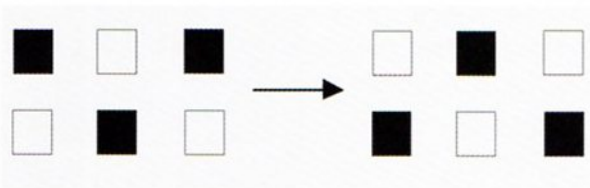
Im Baukasten ist eine CD-ROM enthalten, auf der sich LLWin-Beispielprogramme für die Modelle des Baukastens befinden. Um die Programme öffnen zu können, benötigst du die Software LLWin ab Version 3.0. Du kannst entweder die Beispielprogramme auf der CD lassen und aus LLWin heraus mit dem Befehl DATEI – ÖFFNEN aufrufen oder den kompletten Ordner BIONIC-ROBOTS von der CD in das Projektverzeichnis von LLWin auf die Festplatte kopieren und die Beispiele von dort aus öffnen.

Bevor du die Modelle baust, musst du auch noch einige Teile montieren, z. B. Kabel und Stecker. Was genau zu tun ist, wird in der Bauanleitung beschrieben. So, jetzt ist es so weit. Nun kannst du eintauchen in die faszinierende Welt der laufenden Fischertechnik-Roboter. Sobald du das erste Modell fertig hast und es auf beinahe gespenstische Weise anfängt sich fort zu bewegen, wirst du begeistert sein von dieser Technik, die in der Natur schon seit Millionen von Jahren für die Fortbewegung benutzt wird.

### 3. Laufen auf 6 Beinen

#### 3.1 Gangart der Insekten

Die Gangart der Insekten eignet sich hervorragend als Vorbild für den Antrieb von „maschinellen Sechsheinern“. Beim sogenannten Dreifußgang heben immer drei der sechs Beine gleichzeitig vom Boden ab, das vordere und hintere Bein der einen Seite zusammen mit dem mittleren Bein der anderen Seite:

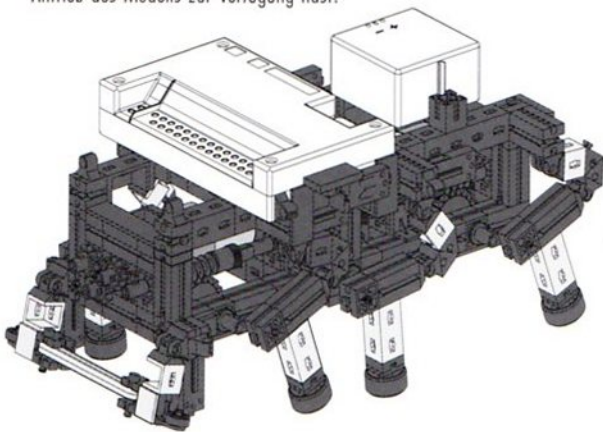


Die Beine, die auf dem Boden stehen (schwarz dargestellt), bilden ein stabiles Dreibein, so dass das Modell immer sicher steht und beim Laufen nicht umkippt.

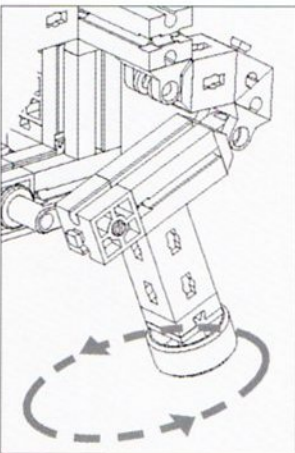
#### 3.2 Modell Mike

##### 3.2.1 Konstruktion des Modells

Baue nun den Sechsheiner Mike (siehe Bauanleitung S. 4). Lade während des Bauens den Akkupack auf, damit du später genügend Energie zum Antrieb des Modells zur Verfügung hast.



Die Beine des Modells sind so konstruiert, dass sie ein sogenanntes Viergelenkgetriebe ergeben. Die Bauform des hier verwendeten Viergelenks nennt man „Kurbelschwinge“. Angetrieben von einer Kurbel führen die beweglich gelagerten Glieder des Getriebes schwingende Bewegungen aus.



Die Abstände zwischen den einzelnen Gelenken und die Lage des Fußpunktes (das ist das untere Ende des Beins), sind so gewählt, dass der Fußpunkt eine elliptische Bewegung beschreibt, wenn sich die Antriebskurbel dreht.

Dadurch entsteht eine Bewegung, die einem Schritt beim Laufen ähnelt. Die 6 Kurbeln, die die Beine antreiben, müssen genau so justiert werden, wie in der Bauanleitung gezeigt. Die drei Beine, die

gleichzeitig auf dem Boden aufsetzen, haben die gleiche Kurbelstellung. Die Kurbeln der 3 Beine, die zu diesem Zeitpunkt in der Luft stehen, sind dazu um 180° verdreht. Die richtige Stellung der Kurbeln zueinander gewährleistet, dass das Modell in der richtigen Schrittfolge, dem Dreifußgang, laufen kann.

Die Zangen- und Nabenmuttern, mit denen man die Schnecken und Zahnräder auf den Achsen fixiert, müssen gut festgedreht werden, damit sich die Kurbeln während des Laufens nicht verstellen.

Die rechte und linke Seite des Modells werden von je einem Motor angetrieben (das wird für das Kurvenlaufen benötigt). Deshalb muss dafür gesorgt werden, dass sich das mittlere Bein der einen Seite immer in der gleichen Stellung befindet wie die beiden äußeren Beine der anderen Seite. Diese Synchronisation erfolgt softwaregesteuert über die Taster E1 und E2.

Teste mit der Interfacediagnose, ob alle Taster und Motoren richtig angeschlossen sind. Drehrichtung der Motoren: linke Drehrichtung = vorwärts

##### 3.2.2 Das erste Programm

Nun beginnen wir damit, Mike etwas beizubringen. Zuerst soll das Modell nur geradeaus laufen. Um das Kurvenlaufen und das Reagieren auf Hindernisse kümmern wir uns später.

###### Aufgabe 1:

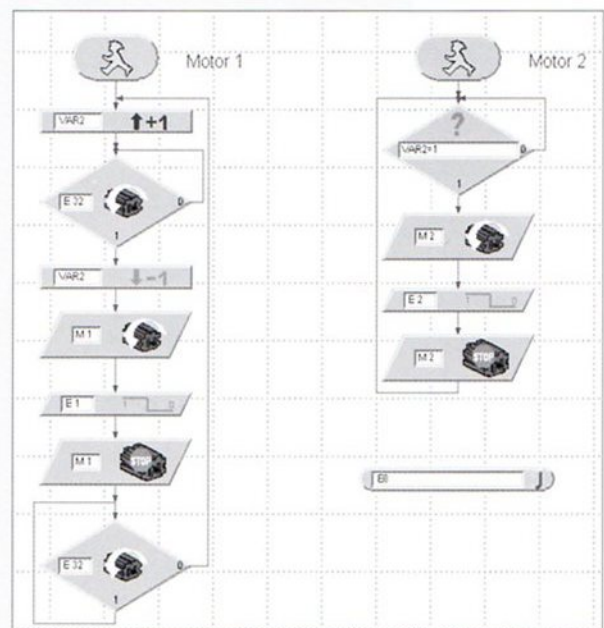
Programmiere das Modell so, dass es im Dreifußgang geradeaus läuft. Benutze die Taster E1 und E2 zur Synchronisation der linken und rechten Beine. Beachte dabei, dass immer die beiden äußeren Beine der einen Seite und das mittlere Bein der anderen Seite die gleiche Stellung haben. Verwende außerdem den Taster E8 als Reset-Taster.

###### Tipps:

Programmiere für jeden Motor einen eigenen Ablauf. Steuere den Ablauf für Motor M2 mit Hilfe einer Variablen VAR2. Wenn du während des Programmierens das Interface nicht benötigst, solltest du die Stromverbindung zwischen Akkupack und Interface unterbrechen um Energie zu sparen.

###### Lösung:

Das Programm für das Geradeauslaufen sieht wie folgt aus:



**D**

Die Variable VAR2 gibt den Impuls, dass Motor M2 startet. Dann wird Motor M1 gestartet. Sobald der Taster E1 betätigt wird, stoppt M1. Sobald E2 betätigt wird, stoppt M2. Der erste Ablauf wartet, bis M2 angehalten hat (Zustand des Motors M2 wird über E32 abgefragt; siehe auch „Abfragen des Motorzustands“ im Handbuch LLWin).

Übrigens, wenn du keine Lust hast diesen Ablauf selbst zu erstellen, findest du ihn als Beiprojekt MIKE\_GERADE.MDL auf der beiliegenden CD. Starte das Projekt. Wenn du alles richtig programmiert hast, kommt jetzt Leben in das Modell und es läuft geradeaus. Herzlichen Glückwunsch. Der erste Schritt ist getan.

### 3.2.3 Die Linksdrehung

Es reicht uns natürlich noch längst nicht, dass Mike nur geradeaus läuft. Als Nächstes wollen wir, dass er sich auf der Stelle dreht.

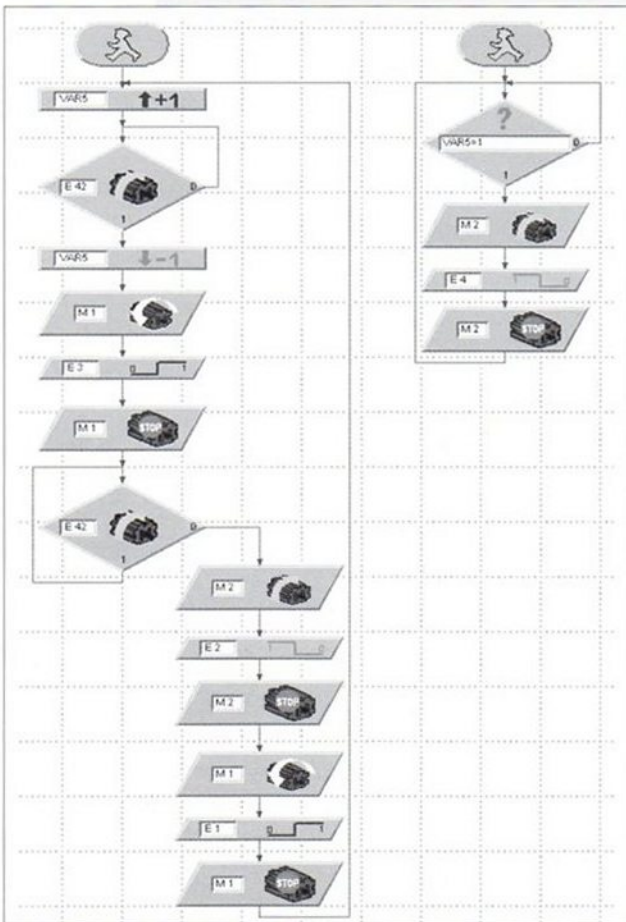
#### Aufgabe 2:

Programmiere Mike so, dass er sich nach links dreht.

#### Tipps:

Das Modell dreht sich nach links, wenn sich M1 nach links und M2 nach rechts dreht.

Du kannst das Modell natürlich unsynchronisiert betreiben. Es dreht sich dann auch, allerdings gibt es dann Stellungen, in denen das Modell nach vorne kippt. Das kann man vermeiden. Und zwar mit folgendem Ablauf:



Mit Hilfe der Taster E1-E4 bewegen sich die linke und die rechte Seite des Modells zuerst einen Schritt gleichzeitig, dann macht die linke Seite einen Schritt, anschließend die rechte usw. So kippt das Modell nie nach vorne. Probiere es aus! Dann fällt es dir auch leichter, diese Reihenfolge nachzuvollziehen.

Auch diesen Ablauf findest du als Projekt MIKE\_LINKS.MDL auf der CD.

Jetzt kann das Modell geradeaus laufen und sich nach links drehen. Es fehlt noch das Rückwärtslaufen und die Rechtsdrehung. Das Rückwärtslaufen funktioniert im Prinzip wie das Vorwärtslaufen, nur mit umgekehrter Motordrehrichtung. Die Rechtsdrehung funktioniert im Prinzip umgekehrt wie die Linksdrehung.

### 3.2.4 Links, rechts, vor, zurück

#### Aufgabe 3:

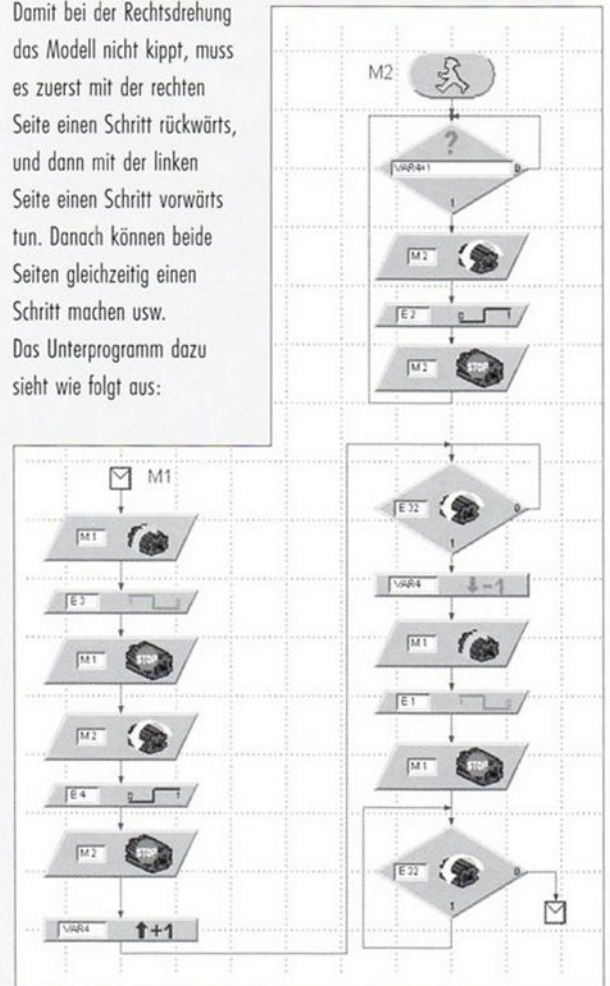
Programmiere nun jede der Funktionen GERADEAUS, ZURÜCK, LINKS und RECHTS als Unterprogramm, damit du sie später in verschiedenen Projekten flexibel einsetzen kannst.

#### Tipps:

Wie man einen vorhandenen Ablauf in ein Unterprogramm kopiert, ist im LLWin-Handbuch beschrieben.

Verwende in jedem Unterprogramm eine andere Variable (VAR2-VAR5) um den Ablauf für Motor M2 zu starten.

Damit bei der Rechtsdrehung das Modell nicht kippt, muss es zuerst mit der rechten Seite einen Schritt rückwärts, und dann mit der linken Seite einen Schritt vorwärts tun. Danach können beide Seiten gleichzeitig einen Schritt machen usw. Das Unterprogramm dazu sieht wie folgt aus:



Die anderen Unterprogramme haben wir an dieser Stelle nicht abgedruckt. Falls du beim Programmieren eines Ablaufs Schwierigkeiten hast, findest du die fertigen Unterprogramme in der Datei MIKE\_VORLAGE.MDL auf der CD. Das Hauptprogramm dieses Projekts ist leer. Im Bausteinfenster unter dem Reiter „Unterprogramme“ findest du dann die Liste mit den vorhandenen Unterprogrammen, die du im Hauptprogramm einfügen kannst.

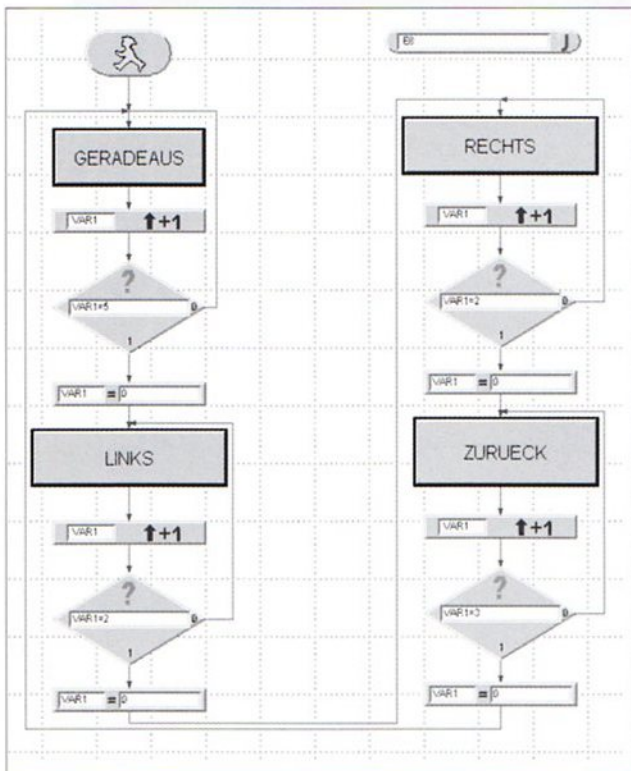


Aber schau nicht gleich dort nach wie es geht. Versuche zunächst einmal selbst auf die Lösung zu kommen. Wenn du es nicht schaffst, kannst du immer noch nachsehen. Um alle Unterprogramme auszuprobieren, wollen wir Mike jetzt tanzen lassen.

**Aufgabe 4:**

Programmiere Mike so, dass er 5 Schritte nach vorne macht, sich 2 Schritte nach links dreht, dann 2 Schritte nach rechts, anschließend 3 Schritte zurück und dann wieder von vorne beginnt. Verwende als Zählvariable für die Anzahl der Schritte die Variable Var1. Benutze E8 als Reset-Taster.

**Lösung:**



Dieses Projekt heißt MIKE\_TANZ.MDL.

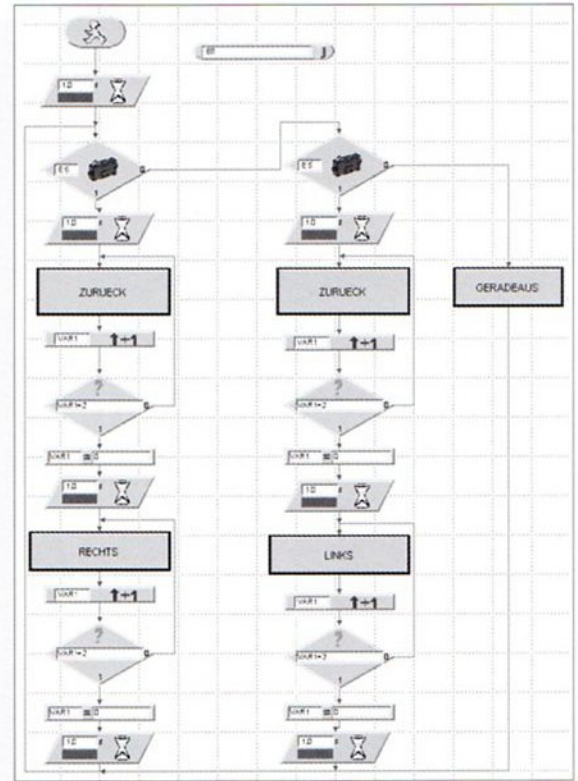
**3.2.5 Hindernisse erkennen**

Zuletzt wollen wir Mike noch dazu bringen, dass er mit seiner beweglichen Stoßstange (oder nennen wir es besser „Fühler“) Hindernisse erkennt und ihnen ausweicht.

**Aufgabe 5a:**

Programmiere Mike so, dass er bei einem Hindernis an seinem linken Fühler (Taster E6) zuerst 4 Schritte zurück und dann 2 Schritte nach rechts ausweicht. Befindet sich ein Hindernis an seinem rechten Fühler (Taster E5) soll er 4 Schritte zurück und dann 3 Schritte nach links ausweichen.

**Lösung:**



Mike läuft zunächst immer geradeaus. Nach jedem Schritt werden die Taster E5 und E6 abgefragt. Ist E6 gedrückt, verzweigt das Programm in den linken Ablauf (erst zurück, dann rechts). Ist E5 gedrückt, geht es in den mittleren Ablauf (erst zurück, dann links).

Da die Taster E5 und E6 nur nach jedem vollen Schritt abgefragt werden, dauert es relativ lange, bis Mike auf ein Hindernis reagiert.

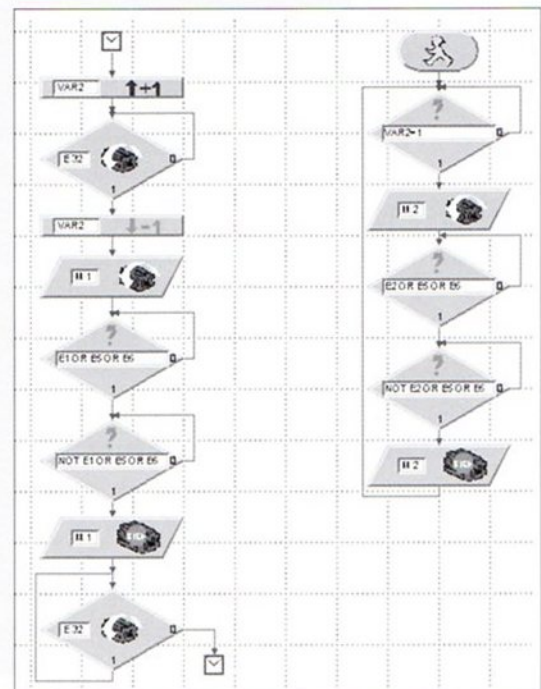
**Aufgabe 5b:**

Optimiere das Unterprogramm GERADEAUS so, dass Mike schneller auf ein Hindernis reagieren kann.

**Tipp:**

Verwende zum Abfragen der Taster E1 und E2 nicht den Baustein FLANKE sondern den Baustein VERGLEICH. Frage damit zusätzlich ab, ob E5 oder E6 gedrückt ist.

**Lösung:**



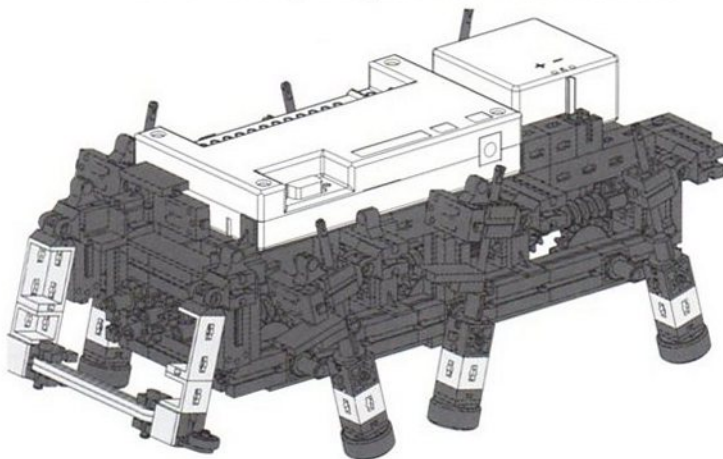
Nun sollte Mike perfekt funktionieren. Dieses Programm befindet sich ebenfalls auf der CD unter MIKE\_HINDERNIS.MDL. Das verbesserte Unterprogramm kannst auch in das Projekt MIKE\_VORLAGE.MDL einarbeiten. Werden in einem anderen Programm E5 und E6 nicht abgefragt, stört das überhaupt nicht. Diese verbesserte Vorlage haben wir unter MIKE\_VORLAGE\_HINDERNIS.MDL abgespeichert.

Nachdem wir nun den ersten Sechsbener ausführlich behandelt haben, wenden wir uns dem zweiten Modell zu, das ebenfalls 6 Beine besitzt. Wir nennen es „Jack“.

## 3.3 Modell Jack

Jack gehört ebenfalls zur Gattung der sechsbeinigen fischertechnik-Modelle. Allerdings unterscheidet er sich in der Konstruktion seiner Beine erheblich von Mike.

Baue nun das Modell wie in der Bauanleitung ab S. 12 beschrieben. Übrigens, die Baustufen 1-13 sind bei Mike und Jack identisch. Du brauchst also Mike nicht völlig zu zerlegen, bevor du anfängst Jack zu bauen.

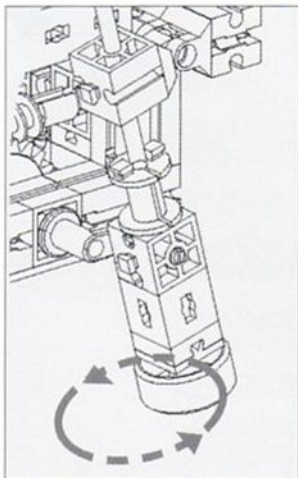


### 3.3.1 Die Konstruktion

Bei der Beinbauweise von Jack handelt es sich ebenfalls um ein sogenanntes Viergelenkgetriebe. Die hier verwendete Bauform nennt man „schwingende Kurbelschleife“. Die Schubstange ist in einer beweglichen Längsführung gelagert, die hin und her schwingt, wenn sich die Kurbel dreht. Die Kurve, die der Fußpunkt des Beins beschreibt, ist nicht so ellipsenförmig wie beim Modell Mike, sondern mehr ein Kreis.

Dadurch hebt und senkt sich Jacks Körper während des Laufens stärker als der von Mike. Die Schritte sind kürzer. Dafür kann Jack kleine Hindernisse überwinden, wozu Mike nicht in der Lage ist. Außerdem erinnert diese Getriebebauform eher an ein Bein als dies bei Mike der Fall ist. Wenn Jack läuft, sieht es aus, als ginge er auf Stelzen.

Er bewegt sich ebenfalls im Dreifußgang der Insekten. Auch bei diesem Modell ist es wichtig, die Kurbeln genau wie in der Bauanleitung beschrieben zu justieren und die Zangen- und Nabenmutter gut fest zu drehen.



### 3.3.2 Die Programmierung

Es liegt nahe zu denken, dass man für Jack die Programme verwenden könnte, mit denen auch Mike funktioniert. Versuche es!

#### Aufgabe 1:

Betriebe Jack mit dem Programm MIKE\_HINDERNIS.MDL. Was kannst du beobachten?

#### Beobachtung:

Vorwärts und rückwärts läuft das Modell einwandfrei. Beim Drehen nach links und rechts kippt es nach vorne.

#### Aufgabe 2:

Wie erklärst du dir das?

#### Lösung:

Die beiden Modelle besitzen unterschiedliche Beinbauweisen. Auch die Taster E1-E4 werden an einer anderen Position des Beins betätigt. Die Art und Weise wie Mike sich dreht, muss demnach für Jack noch lange nicht funktionieren – Pech gehabt.

Selbstverständlich gefällt uns das gar nicht und wir wollen so schnell wie möglich eine Lösung finden.

#### Aufgabe 3:

Versuche Jack so zu programmieren, dass er wie Mike Hindernisse erkennt, aber beim Drehen nicht nach vorne kippt.

#### Tipps:

Speichere das Projekt MIKE\_HINDERNIS.MDL unter dem Namen JACK\_HINDERNIS.MDL und nimm dort die notwendigen Änderungen vor.

Wenn Jack sich dreht, können immer beide Motoren gleichzeitig laufen. Das abwechselnde Ein- und Ausschalten entfällt. Entscheidend ist, dass sich die Beine zu Beginn des Drehens, also nach dem Rückwärtslaufen, in der richtigen Ausgangsposition befinden.

#### Linksrotation:

Soll sich das Modell nach links drehen, muss zu Beginn der Drehung die Kurbel des vorderen linken Beines nach hinten und die des vorderen rechten Beines nach vorne zeigen. Dies ist der Fall, wenn beim Rückwärtslaufen auf der linken Seite der Taster E2 und auf der rechten Seite der Taster E1 betätigt und wieder losgelassen wurde.

#### Rechtsrotation:

Soll sich das Modell nach rechts drehen, muss zu Beginn der Drehung die Kurbel des vorderen linken Beines nach vorne, die des vorderen rechten Beines nach hinten zeigen. Dies ist der Fall, sobald beim Rückwärtslaufen auf der linken Seite der Taster E4 und auf der rechten Seite der Taster E3 betätigt und wieder losgelassen wurde.

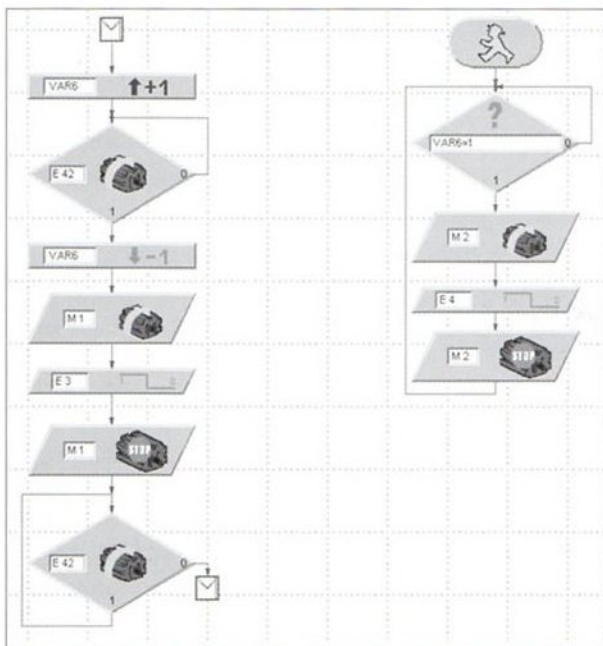
Ziemlich kompliziert, nicht wahr? Aber keine Sorge, wir haben es gleich:

Für die Rückwärtsbewegung müssen zwei unterschiedliche Unterprogramme benutzt werden.

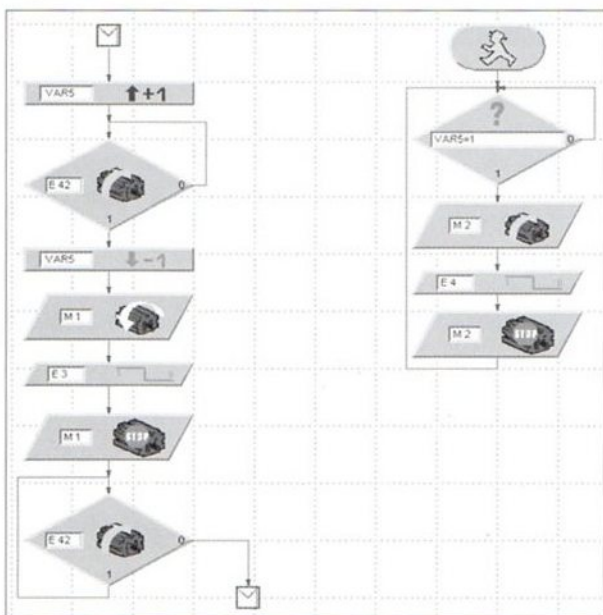
Soll sich das Modell nach links drehen, synchronisierst du die Schritte beim Rückwärtslaufen mit den Tastern E1 und E2. Dies entspricht dem Unterprogramm ZURUECK aus dem Projekt MIKE\_HINDERNIS.MDL.

Soll sich das Modell nach rechts drehen, werden die Schritte rückwärts mit E3 und E4 synchronisiert.

Du benennst also das Unterprogramm ZURUECK mit dem Befehl **UNTERPROGRAMM – UMBENENNEN** in ZURUECK\_L um. Danach kopierst du es mit **UNTERPROGRAMM – KOPIEREN** in ein zweites Unterprogramm ZURUECK\_R. Dort änderst du die Tasterbezeichnungen für die Synchronisierung auf E3 und E4 um. Vergiss auch nicht, für ZURUECK\_R eine neue Variable VAR6 für die Synchronisierung zu benutzen, sonst ist das Chaos perfekt. ZURUECK\_R sieht dann wie folgt aus:

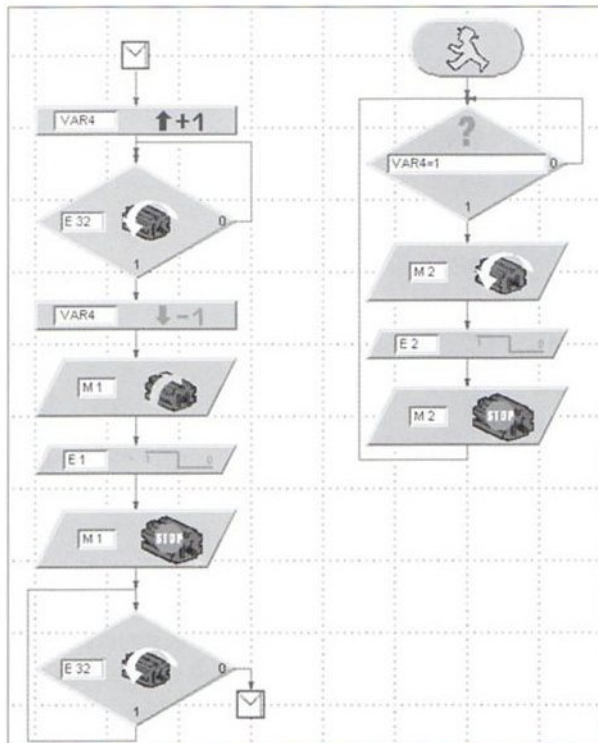


Jetzt müssen noch die Unterprogramme für die Drehung selbst geändert werden, so dass sich die beiden Motoren immer gleichzeitig drehen. Das Unterprogramm LINKS besteht aus folgenden Bausteinen:

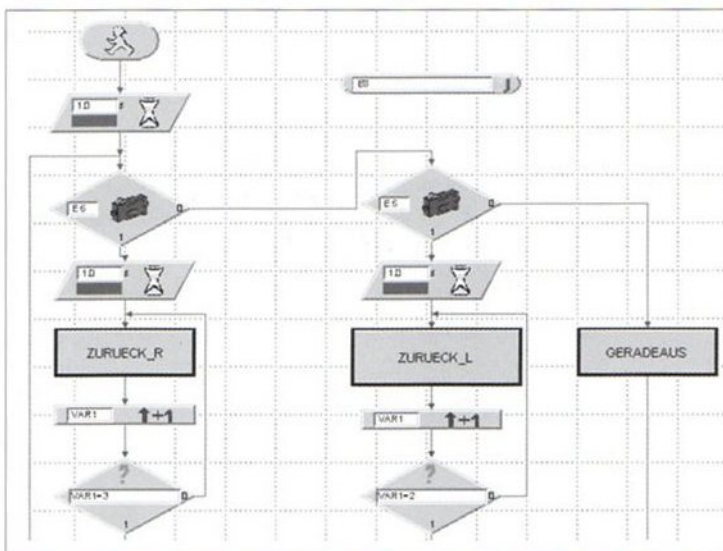


Du siehst, gegenüber dem Unterprogramm im Projekt MIKE\_HINDERNIS.MDL können einige Bausteine entfallen.

Das Unterprogramm für die rechte Drehrichtung sieht ähnlich aus, nur mit anderen Motordrehrichtungen. Außerdem werden die Taster E1 und E2 zur Synchronisierung der beiden Motoren verwendet:



Zuletzt ersetzt du im Hauptprogramm in der Verzweigung für das Ausweichen nach rechts das Unterprogramm ZURUECK\_L durch ZURUECK\_R:



Der Rest des Hauptprogramms bleibt unverändert.

Geschafft! Wenn du nirgends einen Fehler gemacht hast, müsste Jack jetzt laufen ohne beim Drehen zu kippen. Falls irgendetwas nicht funktioniert und du absolut nicht weißt warum, mach dir nichts draus, das war auch wirklich eine harte Nuss. Du hast auf jeden Fall die Möglichkeit, das fertige Projekt JACK\_HINDERNIS.MDL einfach von der CD aufzurufen und das Modell damit zu betreiben.

Falls du es selbst geschafft hast das Problem zu lösen, kannst du stolz sein, denn dann gehörst du ab jetzt zu den Profis unter den Programmierern.

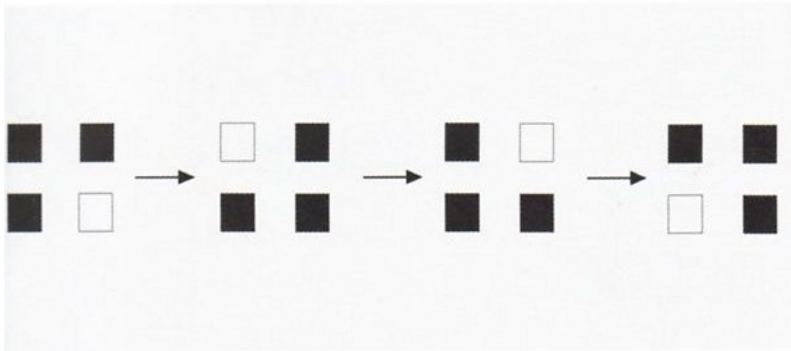


## 4. Laufen auf 4 Beinen

### 4.1 Gangarten der Säugetiere

Um einen Laufroboter mit 4 Beinen zu konstruieren, nehmen wir wieder die Natur zum Vorbild und schauen uns an, in welchen Gangarten sich Säugetiere fort bewegen.

Die langsamste und sicherste Gangart ist der sogenannte Schritt. Ein Bein sucht einen neuen Standort, während sich der Körper des Tieres auf drei Beine stützt. Die Tiere bewegen sich im Kreuzgang mit der Schrittfolge: rechtes Vorderbein, linkes Hinterbein, linkes Vorderbein, rechtes Hinterbein vorwärts.

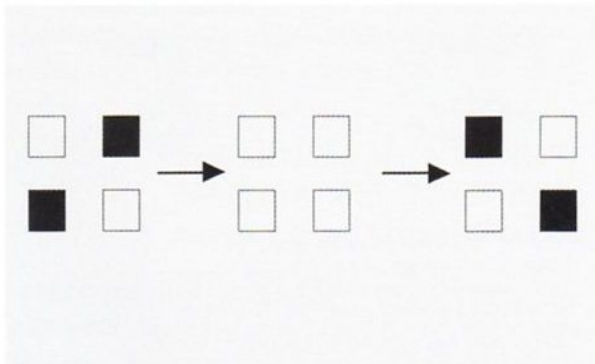


Die schwarzen Flächen stellen die Beine dar, die auf dem Boden stehen, die weißen Flächen das abgehobene Bein.

Wollen wir diese Gangart für einen Laufroboter verwenden, müssen wir folgendes bedenken:

Stell dir vor, du sägst an einem Tisch mit 4 Beinen ein Bein ab. Was passiert? Richtig, der Tisch kippt um. Die drei Beine bilden also kein stabiles Dreibein mehr, wie das bei den Sechsheinern der Fall war. Das erschwert die Konstruktion eines vierbeinigen Roboters.

Je schneller sich die Säugetiere fort bewegen, desto instabiler wird ihre Gangart. Sehen wir uns noch kurz die Gangart „Trab“ an. Im Trab werden die Beine in der Diagonalen synchron abgehoben. Bevor sie aber den Boden berühren springen die beiden anderen Beine bereits ab. Dies bedeutet, dass zeitweise der Bodenkontakt ganz verloren geht.

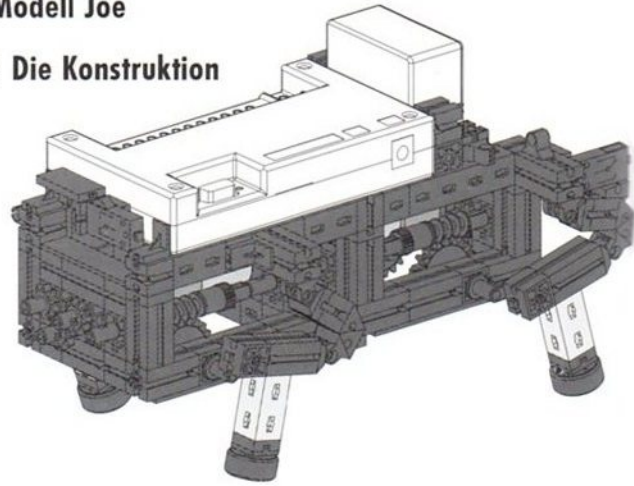


Du kannst dir sicher vorstellen, dass eine Gangart, bei der der Bodenkontakt zeitweise verloren geht, für ein fischertechnik-Modell, auf dem sich das Interface und der Akkupack befinden sollen, nicht unbedingt geeignet ist.

Versuchen wir es also mit der Gangart „Schritt“.

## 4.2 Modell Joe

### 4.2.1 Die Konstruktion



Baue das Modell so zusammen, wie es in der Bauanleitung ab S. 20 beschrieben ist.

Die Konstruktion der Beine ist gleich wie bei Mike. Die Stellung der Kurbeln, die die Beine antreiben, muss bei Joe völlig anders sein. Die Kurbeln sind zueinander um jeweils 90° versetzt. Du musst sie genau so justieren, wie in der Bauanleitung angegeben. Die Synchronisierung der linken und rechten Seite erfolgt wieder über die beiden Taster E1 und E2. Damit erhalten wir die benötigte Schrittfolge.

Damit uns das Modell nicht umkippt, sobald ein Bein angehoben wird, muss der Schwerpunkt des Modells so liegen, dass das Modell im richtigen Augenblick kippt und von dem jeweils gerade entlasteten Bein aufgefangen wird.

### 4.2.2 Die Programmierung

Bei diesem Modell wollen wir uns mit dem Geradeauslauf begnügen.

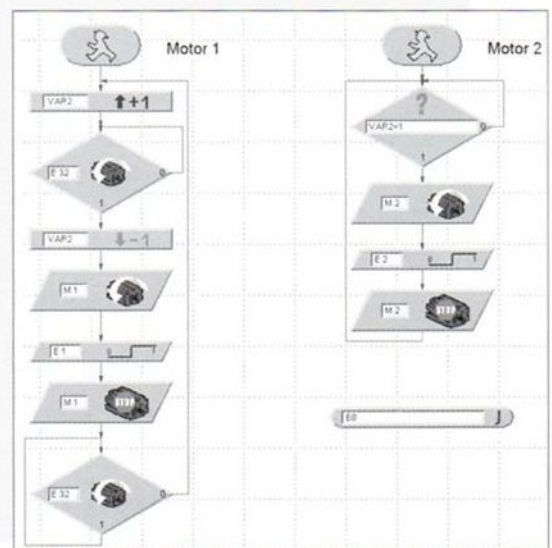
#### Aufgabe 1:

Programmiere Joe so, dass er sich in der Gangart „Schritt“ vorwärts bewegt.

#### Tipps:

Verwende für jeden Motor einen getrennten Ablauf und synchronisiere die beiden Seiten mit den Tastern E1 und E2. Benutze E8 als Reset-Taster.

#### Lösung:



Bei diesem Programm benutzen wir die 0-1 Flanke der Taster für die Synchronisierung. In dem Moment, wenn die Taster gedrückt sind, haben die Kurbeln aller 4 Beine die richtige Stellung zueinander. Das Projekt nennen wir JOE.MDL.

Du siehst, dass sich Joe weitaus schwerfälliger bewegt als Mike und Jack. Durch die notwendige Gewichtsverlagerung schwankt der Körper ziemlich stark und der Laufstil ist bei weitem nicht so elegant wie bei den Sechsbeynern.

Falls du noch Lust hast, diesem Modell das Kurvenlaufen beizubringen, versuche einfach, ob du es schaffst. Viel Glück dabei.

## 5. Laufen auf zwei Beinen

### 5.1 Zweibeinige Läufer

Das Laufen auf zwei Beinen ist nicht in der Gattung der Säugetiere entstanden, sondern wird auch von einigen Reptilienarten praktiziert. Warane, Leguane, Agamen und Rennechsen benutzen auf der Flucht nur ihre Hinterbeine. So erreichen sie sehr große Schrittweiten und werden dadurch unheimlich schnell. Sie benötigen dazu kräftige Hinterbeine, einen langen Balancierschwanz und ebenes Gelände.

Vögel gehören ebenfalls zu den Zweibeinern. Zu den schnellsten Laufvögeln gehört der Strauß. Er erzielt Dauergeschwindigkeiten von bis zu 60 km/h.

Den perfektesten Zweibeiner stellt der Mensch dar. Der völlig aufrechte Gang verlangt die Streckung des Hüftgelenks. Diese wird durch den großen Gesäßmuskel gewährleistet. Außerdem können die Beine im Kniegelenk „eingearastet“ werden und so in einer energiearmen Haltung fixiert werden.

Die Fortbewegung auf zwei Beinen stellt die schwierigste aller Gangarten dar, denn sie erfordert neben den beschriebenen anatomischen Voraussetzungen einen ausgeprägten Gleichgewichtssinn. Uns Menschen erscheint das Gehen auf zwei Beinen selbstverständlich und einfach. Doch wenn wir uns vor Augen führen, dass beim Abheben eines Beines der ganze Körper nur auf einem Bein ruht und so ausbalanciert werden muss, erkennen wir, dass gerade das Halten des Gleichgewichts diese Fortbewegungsart so kompliziert gestaltet. Selbst ein neugeborener Mensch ist nicht sofort in der Lage auf zwei Beinen zu gehen. Er krabbelt zuerst „auf allen Vieren“, bevor er sich aufrichtet und „Laufen lernt“.

An der Waseda-Universität in Tokio wurden bereits zweibeinige Roboter entwickelt, die sich mit Hilfe zahlreicher Gelenke, verschiedenster Sensoren, Kameras und leistungsfähiger Mikroprozessoren bewegen und durch Gewichtsverlagerung das Gleichgewicht halten.

Für unseren Baukasten Bionic Robots jedoch wäre das zu aufwendig und zu kompliziert. Wir haben gesehen, dass wir bereits beim Laufen auf vier Beinen mit einem fischertechnik-Modell langsam an unsere Grenzen stoßen.

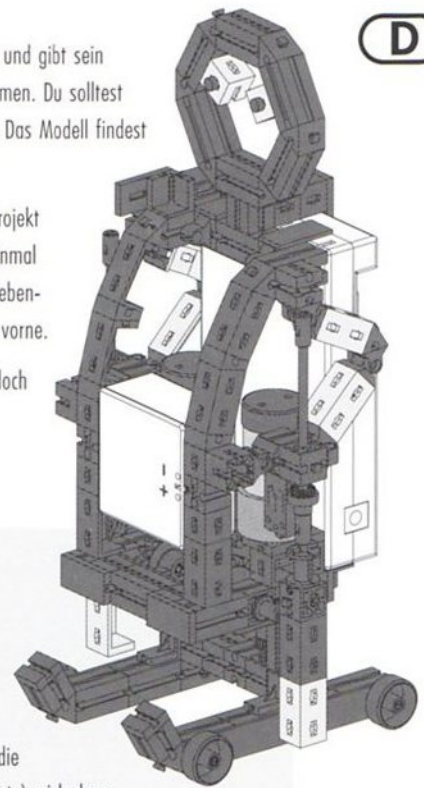
### 5.2 Modell Jim

Damit wir dieses Kapitel aber nicht nur theoretisch betrachten, haben wir uns zum Schluss noch erlaubt, wenigstens einen zweibeinigen Skifahrer, wir nennen ihn Jim, zu konstruieren. Er hat zwar wenig mit einem zweibeinigen

Läufer zu tun, ist aber unheimlich nett und gibt sein Bestes, um irgendwie vorwärts zu kommen. Du solltest dir diesen Spaß nicht entgehen lassen. Das Modell findest du in der Bauanleitung auf S. 27.

Als Programm kannst du einfach das Projekt JOE.MDL verwenden. Du musst nicht einmal was daran verändern. Jim funktioniert ebenfalls damit und stochert langsam nach vorne.

Eine Aufgabe wollen wir dir nun aber doch noch stellen:



#### Aufgabe 1:

Programmiere Jim so, dass er ca. 50 cm vorwärts läuft, sich dann nach rechts um 180° dreht, die gleiche Strecke zurück läuft (vorwärts), sich dann 180° nach links dreht, wieder die gleiche Strecke läuft usw. Verwende für die Anzahl der Schritte geradeaus den Terminalparameter EA, für die Anzahl der Schritte links EB und für rechts EC. Verwende wieder E8 als Reset-Taster.

#### Tipps:

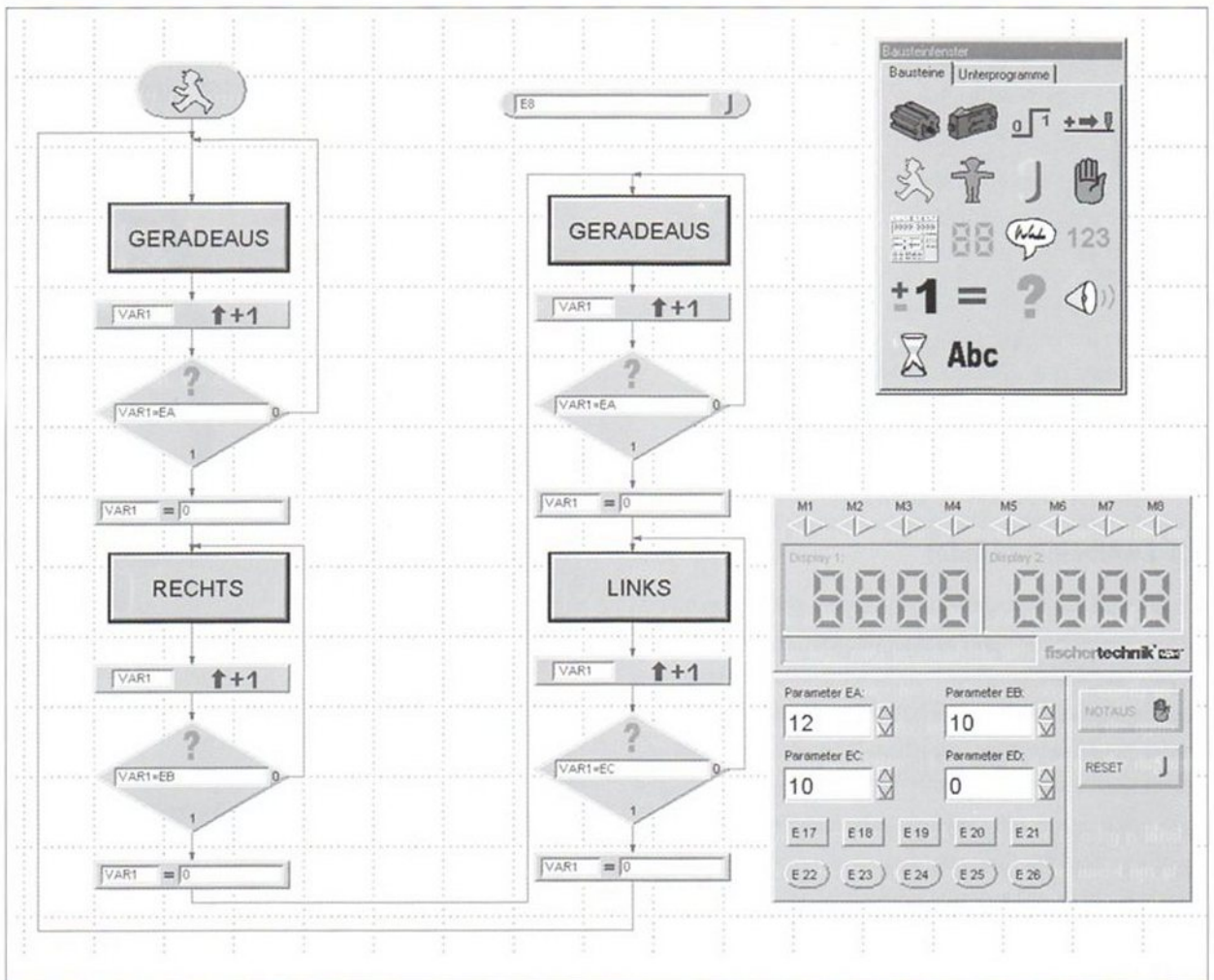
Speichere das Projekt JOE.MDL unter JIM.MDL ab. Mache darin aus dem Hauptprogramm ein Unterprogramm „Geradeaus“ (Bausteine markieren und ausschneiden, über BEARBEITEN – UNTERPROGRAMM neues Unterprogramm erstellen, Bausteine einfügen, SUBIN und SUBOUT ergänzen, siehe auch LLWin-Handbuch).

Erstelle aus diesem Unterprogramm mit dem Befehl UNTERPROGRAMM – KOPIEREN die benötigten Unterprogramme LINKS und RECHTS. Ändere darin die Motordrehrichtung und Abfrage der Motordrehrichtungen entsprechend ab und verwende für jedes Unterprogramm eine andere Steuervariable für Motor M2.

Danach programmierst du das Hauptprogramm ähnlich wie bei MIKE\_TANZ.MDL. Nur dass du für die Anzahl der Schritte die einstellbaren Terminalparameter EA-EC verwendest. Wie viele Schritte Jim benötigt um sich um 180° zu drehen bzw. einen halben Meter vorwärts zu kommen, musst du ausprobieren.

#### Lösung:

Nachfolgend bilden wir das Hauptprogramm ab. Die Unterprogramme kannst du dir falls nötig wieder direkt am Bildschirm ansehen. Auch bei uns heißt das Projekt JIM.MDL.



Wir haben in dem Projekt gleich noch das Unterprogramm ZURUECK ergänzt, auch wenn es hier direkt nicht benötigt wird. Aber bestimmt willst du, dass Jim noch andere Wege geht. Vielleicht muss er sich dabei ja auch einmal rückwärts bewegen.

## 6. Zusammenfassung

Auf deiner Reise durch die Welt der Bionic Robots von fischertechnik hast du sicher festgestellt, dass es nicht immer ganz leicht war, die vier Jungs zum Laufen zu bringen. Es ist eben schon schwieriger, sich auf Beinen fort zu bewegen, als einfach auf Rädern zu rollen. Besonders das Programmieren der Synchronisierung zwischen der linken und rechten Seite beim Drehen nach links oder rechts erfordert etwas Konzentration. Aber für diejenigen, die sowieso mehr Spaß am Bauen der Modelle finden, haben wir ja alle Programme fix und fertig auf die CD gebrannt, so dass jeder die Modelle bauen und betreiben kann.

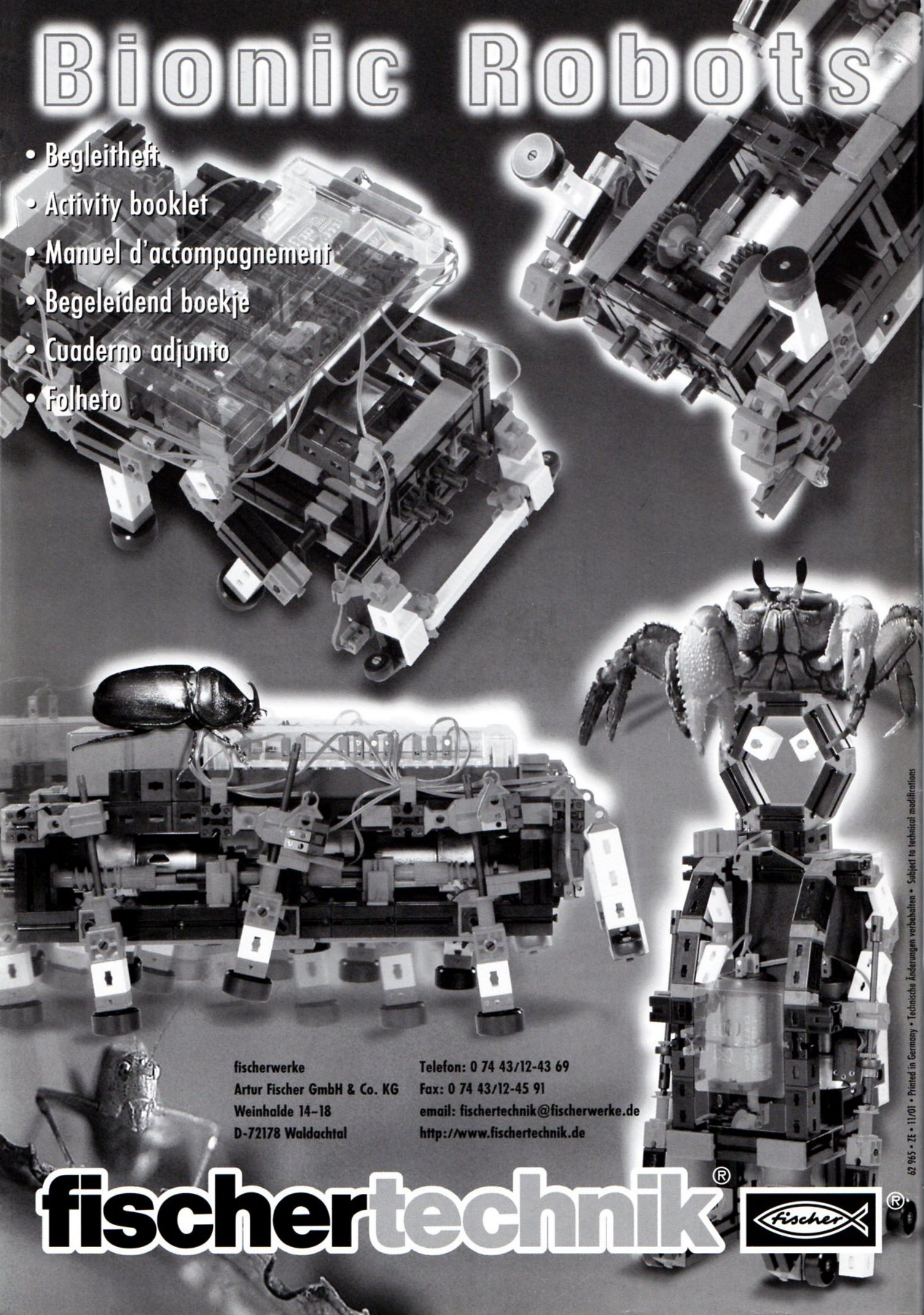
Falls du zu den Profi-Programmierern gehörst, hast du sicherlich noch viele Ideen, welche Aufgaben man für Mike, Jack, Joe oder Jim noch programmieren kann, sei es mit zusätzlichen Sensoren, damit sie nicht vom Tisch fallen, oder dass sie sich in einem Labyrinth zurecht finden.

Mit zusätzlichen Bauteilen kannst du sie auch noch mit einem Kopf, Rüssel, oder Schwanz ausstatten. Deiner Phantasie sind dabei keine Grenzen gesetzt. Lass dir was einfallen!



# 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](mailto:fischertechnik@fischerwerke.de)  
<http://www.fischertechnik.de>

# fischertechnik®

