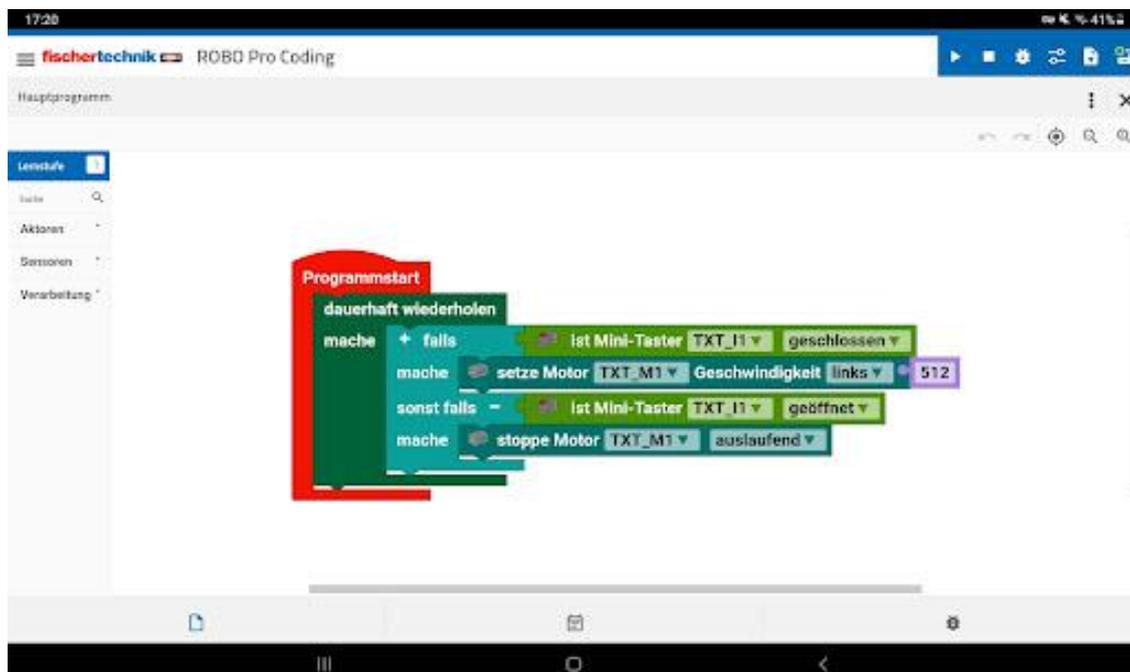


Überblick der Beispielprogramme von ROBO Pro Coding für den TXT 4.0, den RX und den BT Smart Controller

Wer sich erstmalig mit dem neuen Controller TXT 4.0 beschäftigt, vermisst eine zusammenhängende Bedienungsanleitung. Was es aber gibt, sind zahlreiche Beispielprogramme und einige PDF-Dateien auf der Seite von Fischertechnik zur Erklärung. Ich möchte hier einen Überblick über den Zusammenhang zwischen den Baukästen und den Beispielprogrammen unter Robo PRO Coding darstellen.

Inhalt:

- 1 Aufruf der Beispielprogramme
- 2 Die einzelnen Baukästen
- 3 TXT 4.0 Controller
- 4 Robotics Hightech
- 5 ROBOTICS Base
- 6 Autonomous driving
- 7 Competition
- 8 IoT
- 9 Omniwheels
- 10 Künstliche Intelligenz
- 11 Industrial Robots
- 12 Coding Competition
- 13 RX-Controller
- 14 Smart Robots Max
- 15 Smart Robots Pro
- 16 BT-Smart Beginner Set
- 17 STEM Coding RoboMission



1. Aufruf der Beispielprogramme

Die Beispielprogramme werden über das Menü (1), Neu (2), Beispiele (3), NÄCHSTE (4) aufgerufen. Nach der Auswahl des Baukastens (5) können die Beispielprogramme (6) geladen werden.



Voreinstellungen

> Einstellungen

Projekt

> Neu 2

Einen Baukasten auswählen

TOY Robotics Hightech



ABBRECHEN

VORHERIGE

NÄCHSTE

Eine Vorlage für dein neues Projekt auswählen



Leer



Kamera



Anzeige



Kamera + Anzeige



Beispiel

4

ABBRECHEN

NÄCHSTE

Ein Beispiel für dein neues Projekt auswählen

ServoCalib_DPS



6

ABBRECHEN

VORHERIGE

NÄCHSTE

ERSTELLEN

Mit der Einführung der neuen Programmierumgebung ROBO Pro Coding für den Controller TXT 4.0 wurde der Baukasten „Robotics Hightech“ auf den Markt gebracht. Aufgrund von Lieferschwierigkeiten wurde dieser Kasten zeitweise mit dem alten Controller TXT unter dem Namen „Robotics Smarttech“ ausgeliefert. Um einen neuen Controller TXT 4.0 zu erwerben gibt es außer dem o.g. Baukasten, noch die Möglichkeit diesen separat (560166) zu kaufen. Die dritte Möglichkeit ist der Erwerb des Baukastens „Robotics TXT 4.0 Base Set“ (559888). Darauf aufbauend gibt es eine Vielzahl von Add On-Kästen, die ständig erweitert werden. Mit dem Controller RX (191232) im Kasten 564111, gibt es nun auch die abgespeckte Version des TXT 4.0. Dabei fehlen die Servo-Schnittstellen, das Display und die schnellen Zählgänge. Seit kurzem ist es ebenfalls auch möglich, den älteren Controller BT-Smart mit ROBO Pro Coding zu programmieren.

2. Die einzelnen Baukästen

Grundbaukasten Robotics Hightech

Highend Robotics Produkt



Robotics TXT 4.0 Base Set

Grundlagen der Robotik begreifbar machen.



Add On: Autonomous Driving

Zusatzset für autonomes Fahren zum Robotics Base Set



Add On: Omniwheels

Zusatzset für Omniwheels zum Robotics Base Set



Add On: IoT

Zusatzset für IoT zum Robotics Base Set



Add On: Competition

Zusatzset für Wettbewerbe zum Robotics Base Set



Add On: Add On: Künstliche Intelligenz

Zusatzset für zum Kennenlernen Funktionsweise der KI



Add On: Industrial Robots

Zusatzset für zwei realistische sechsachsige Robotermodelle



Coding Competition

Die Basis zur erfolgreichen Wettbewerbsteilnahme



Smart Robots Max

Experimentierkasten für Einsteiger



Smart Robots Pro

Das Einsteiger-Komplettpaket



BT-Smart Beginner Set

Nun auch für Robo Pro Coding



STEM Coding RoboMission

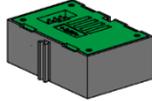
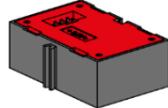
Entwickelt für Geschwindigkeit, Präzision und Höchstleistung, RoboMission der World Robot Olympiad



3.TXT 4.0 Controller

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/560166-txt-4-0-controller>



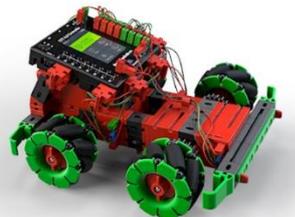
Name d. Datei in RoboPro Coding	Beschreibung	Bild
test_environmental_sensor	Test_Umweltsensor	
test_combi_sensor	Test_Kombisensor	
test_gesture_sensor	Test_Gestiksensor	
test_image_processing Python-Code	Test_Bildverarbeitung	
voltage_extended_range_I1 Python-Code	Spannung_Erweiterter_Bereich_I1	
test_voice	Test_Stimme	
python_programming_template	Python_Programmiervorlage	
python_programming_example	Python_Programmierbeispiel	
test_battery Python-Code	Test_Akku	
test_speedtest Python-Code	Schnelligkeitstest	

Python-Code In diesen Programmen sind Python-Code-Elemente eingesetzt, die noch nicht durch ft dokumentiert sind.

4. Robotics Hightech

<https://www.fischertechnik.de/de-de/produkte/spielzeug/robotics/559895-robotics-hightech>



Name d. Datei in RoboPro Coding	Beschreibung	Bild und PDF-Datei
function_motor_on	Funktion Motor ein	 Funktionsmodell.pdf
function_lightbarrier_motor_on	Funktion Lichtschranke Motor ein	
function_ultrasonic_motor_on	Funktion Ultraschall Motor ein	
function_ultrasonic_motor_speed	Funktion Ultraschall Motordrehzahl	
function_encodermotor_distance	Funktion Encodermotor Entfernung	
cameraman_detect_color	Es sollen 3 verschiedene Farben (rot, gelb, grün) erkannt werden (Farbtafeln). Die Farbe, die erkannt wurde, soll im Display der TXT 4.0 Controllers angezeigt werden.	 Kameramann.pdf
cameraman_alarm	Bewegung im Kamerabild soll erkannt werden. Ist dies der Fall, wird ein akustisches und ein optisches Signal ausgegeben.	
barcodereader_code39	Über die Kamera werden 5 Linien nach dem Prinzip „Barcode 39“ erkannt und in Buchstaben oder Zahlen auf dem Display des TXT 4.0 Controllers angezeigt.	 Barcodeleser.pdf
omniwheels_x2_forward_backward_time	Das Fahrzeug soll eine bestimmte Zeit geradeaus nach vorne und wieder zurückfahren. Anschließend soll das Programm stoppen.	 OmniwheelX2.pdf
omniwheels_x2_forward_backward_dist	Erweitert das Programm um eine impulsgesteuerte Rückfahrt.	
omniwheels_x2_forward_rot90_dist	Der Ablauf erhält zusätzlich eine 90 Grad Drehung.	
omniwheels_x2_square	In dieser Aufgabenstellung soll das Fahrzeug ein Rechteck abfahren.	
omniwheels_x2_linefollower	Fahrt auf der Vorlage als Linienverfolger	
omniwheels_x4_move	Bewegungstest	
omniwheels_x4_forward_backward_time	Sas Fahrzeug soll eine bestimmte Zeit geradeaus nach vorne und wieder zurückfahren. Anschließend soll das Programm stoppen.	
omniwheels_x4_forward_backward_dist	D das Fahrzeug soll über eine bestimmte Distanz (impulsgesteuert) bewegt werden.	
omniwheels_x4_forward_rot90_dist	Der Ablauf erhält zusätzlich eine 90 Grad Drehung.	
omniwheels_x4_forward_diagonal_right_forward	Diagonalfahrt	 OmniwheelX4.pdf
omniwheels_x4_forward_sideways_right / left	Fahrt vorwärts, seitwärts rechts / links	

Name d. Datei in RoboPro Coding	Beschreibung	Bild und PDF-Datei
omniwheels_x4_sensor_linefollower	Einer Spur folgen mit Hilfe des IR-Spursensors.	 OmniwheelX4sensoren.pdf
omniwheels_x4_sensor_inefellerollower ultrasonic	Eine Spur suchen, der Spur folgen und Hindernisse mit dem Ultraschallsensor erkennen.	
omniwheels_x4_sensor_linefollower ultrasonic_acc	Die Entfernung zum Hindernis wird gemessen und die Geschwindigkeit entsprechend dem Abstand angepasst wird.	
omniwheels_x4_sensor_servomove	Testprogramm für die Kamera in Verbindung mit dem Servomotor.	
omniwheels_x4_sensor_linefollower camera_trafficsign	Farberkennung mit der Kamera, Anzeige der Farbflächen auf dem Display, seitliches Einparken, warten am Zebrastreifen.	
omniwheels_x4_sensor_inefellerollower camera_pd	Spurerkennung mit Kamera und PD-Regler. Modell soll an der schwarzen Line des Parcours entlangfahren.	
omniwheels_x4_cam_move	Kamerabewegung	 Omniwheel_x4_ballroboter.pdf
omniwheels_x4_ball_simple	Ball laden und über den Servo feuern	
omniwheels_x4_ball_gui	Seitlich fahren, drehen, Ball laden und abschießen unter Verwendung von Buttons auf dem Display	
omniwheels_x4_target_detect	Ziele (rote, gelbe oder grüne Tafeln) erkennen, Roboter am Ziel ausrichten. Die Auswahl der Farbe wird über die Buttons eingegeben. Roboter sucht das ausgewählte Ziel.	
omniwheels_x4_target_balldetect	Rote, gelbe und grüne Tafeln werden erkannt und abgeschossen – alle drei nacheinander	
omniwheels_x4_ball_voice_control	Fernsteuerprogramm mit Spracheingabe – Sprachsteuerung der Fahrbefehle sowie des Abschusses	
omniwheels_x4_soccer_ballfollower	Das Fußballfeld wird ohne Tor aufgebaut. Der Roboter soll sich im Kreis drehen und einen Ball suchen. Liegt der Ball im Schussbereich der Lichtschranke, wird er weggeschossen.	 Fussballroboter.pdf
omniwheels_x4_soccer_goal	Auf dem Fußballfeld wird ein Tor aufgebaut. Dieses und der Ball wird vom Roboter mit der Kamera erfasst und der Ball wird ins Tor geschossen.	
omniwheels_x4_paint	Über das Display sowie mit der Sprachsteuerung können verschiedene Figuren gezeichnet werden	 Malroboter.pdf

Hinweis: Die Dateien sind nicht in dieser Reihenfolge aufrufbar.

Im Folgenden, werden die Programme der Grundlagen- und Erweiterungsbaukästen aufgezeigt.

Python-Code

Programme mit diesem Symbol arbeiten auch mit Python-Code.



Programme mit diesem Symbol benötigen die Einrichtung einer ft-Cloud.

(A1/A1)

Programme mit dieser Erweiterung sind in einer zusätzlichen Datei beschrieben.

Diese können unter: <https://www.fischertechnik.de/de-de/schulen/lernmaterial/sekundarstufe-programmieren/robotics-base-set-und-add-ons> als Zip-Datei geladen werden

Unterrichtspläne ROBOTICS Base Set →	Unterrichtspläne ADD ON Omniwheels →	Unterrichtspläne Autonomous Driving →	Unterrichtspläne ADD ON IoT →
---	---	--	-------------------------------------

(A1...

Name der Datei

.../A1)

Nummer der Aufgabe

.../E1)

oder Nummer des Experimentes

5.ROBOTICS Base

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/559888-robotics-txt-4-0-base-set>



Barcode_Decoder	9-Stelliger Barcode (A4/A2)
Distance_Measuring	Abstandsmessung (A1/A3)
Resistivity_Survey	Widerstandsmessung (A1/A1)
Resistivity_Survey_with_Color_Code_Carbon_Film_Resistor	Bestimmung Widerstand-Farbcode (3 Farbringe) (A1/E2a)
Resistivity_Survey_with_Color_Code_Metal_Film_Resistor	Bestimmung Widerstand-Farbcode für Metallfilmwiderstände (4 Farbringe) (A1/E2b)
Temperature_Measuring	Temperaturmessung (A1/A2)
Temperature_Measuring_SteinhartHart	Temperaturmessung nach Steinhart-Hart (A1/E1)
Temperature_Measuring_with_Scale	Temperaturmessung mit Display-Scala (A1/A2b)
Temperature_Measuring_with_Scale_and_MinMax	Temperaturmessung mit Display-Scala mit "min" und „max“ Wert (A1/A2c)
Boom_Gate_with_Encoder	Schranke mit Magnetdecoder (A2/A3)
Boom_Gate_with_Pulse_Wheel	Schranke mit Impulsrad (A2/A2)
Boom_Gate_with_Pushbutton_Switch	Schranke mit Bedarfstaster (A2/E1)
Boom_Gate_with_Pushbutton_Switch_and_Lightbeam	Schranke mit Durchfahrtskontrolle (A2/E2)
Boom_Gate_with_Pushbutton_Switch_Lightbeam_and_Counter	Schranke mit Durchfahrt-Zähler (A2/E3)
Boom_Gate_with_Stop_Position_Switch	Schranke mit Endlagentaster (A2/A1)
Carpark_Monitoring_with_two_Boom_Gates	Schranke mit Belegungskontrolle (A2/E4)
Blinking_Light	Blinklicht (A3/E1)
Boom_Gate_with_Pushbutton_Switch_Lightbeam_and_Traffic_Light	Parkhausschranke mit Ampelsteuerung (A3/A2)
Boom_Gate_with_Pushbutton_Switch_Lightbeam_Counter_and_Blinking_Light	Parkhausschranke mit Durchfahrt-Zähler und Blinklicht (A3/E2)
Carpark_Monitoring_with_two_Boom_Gates_and_Blinking_Lights	Parkhausschranke mit Belegungskontrolle und Blinklicht (A3/E3)
Traffic_Light_with_Pushbutton_Switch	Ampelsteuerung (A3/A1)

Bar_Width_Recognition	Balkenerkennung (A4/A1)
Camera_Color_Decoder	Farberkennung (A4/E2)
Code39_Decoder	Code-39-Dekodierer (A4/A3)
Code39_Decoder_with_Error_Detection	Fehlererkennung (A4/A4)
Code39_Template	Code-39-Dekodierer (modifiziert) (A4/A3)
Colored_Barcode_Decoder	Mehrdimensionaler Code (A4/E3)
Line_Detection	Linienenerkennung (A4/A1)
RGB_Colorcode	Farberkennung (A1/E1)
RGB_Decoder	RGB-Code (A4/E1)
Bit_Sequence_Sender	Binärcode-Sender (A5/A2)
Bit_Sequence_Sender_Receive_LED	Bitfolgesender mit Empfangsanzeige (A5/A2c)
Bit_Sequence_Sender_Receiver	Bitfolgeempfänger (A5/E1)
Light_Signal	Lichtsignalempfänger (A5/A1b)
Light_Signal_Counter	Lichtsignalempfänger mit Zähler (A5/A1c)
Morse_Code_Sender	Morde-Zeichen-Sender (A5/A3)
Morse_Code_Sender_Receiver	Morde-Zeichen-Empfänger (A5/E2)
Morse_Code_Sender_Receiver_with_Caesar_Encryption	Verschlüsselung-Caesar-Chiffre (A5/E4)
Morse_Code_Sender_Receiver_with_Vigenere_Encryption	Verschlüsselung-Vigenère-Chiffre (A5/E5)
Morsecode_Template	Programmfragment für Morsecode
Buggy_Driving_Test	Einfache Geradeausfahrt (A6/A1)
Buggy_Driving_Test_Synchronous	Synchroner Antrieb (A6/A2)
Buggy_Driving_Test_Synchronous_Distance	Kontrollierte Fahrt mit Encodern (A6/A3)
Buggy_Synchronous_Turning_Calibration	Drehen um vorgegebenen Winkel (A6/A5)
Buggy_Target_Reacher	Buggy mit Encoder-Navigation (A6/E3)
Buggy_Turning_Test_Synchronous	Synchrones Drehen auf der Stelle (A6/A4)
Buggy_Turning_Test_Synchronous_Angle	Drehen um vorgegebenen Winkel (A6/A5)
Buggy_with_Pushbuttons	Buggy mit Hinderniserkennung (A6/A6)
Buggy_with_Pushbuttons_and_IR_Sensors	Buggy mit Linien-/Kantenerkennung (A6/A7)
Buggy_with_Pushbuttons_IR_Sensors_and_Light	Buggy mit Frontstrahler (A6/E1)
Buggy_with_Ultrasonic	Hinderniserkennung mit Ultraschall (A6/E2a)
Buggy_with_Ultrasonic_and_Light	Ergänzung um Frontstrahler (A6/E2b)
Buggy_with_Ultrasonic_flexible_Turn	Drehung, bis kein Hindernis bei 25 cm (A6/E2c)
Buggy_with_Ultrasonic_flexible_Turn_display_Angle	Drehung mit Winkelanzeige (A6/E2d)
Buggy_Coordinates_Drawing	Malen nach Zahlen (A7/E2)
Buggy_Drawing_Polygons	Zeichnen eines N-Ecks (A7/A3)
Buggy_Driving_Functions	Steuerungsfunktion (A7/A1)
Buggy_House_of_Santa_Claus	Haus vom Nikolaus (A7/A2)
Buggy_Move_2_Point	Zielpunkt ansteuern (A7/E1)
Drawing_Coordinates	Zeichnungs-Koordinaten
Buggy_Line_Follower_digital	Spurfolger mit Spursensor (A8/A1)
Buggy_Line_Follower_digital_optimized	Optimierter Spurfolger (A8/A2)
Buggy_Line_Follower_with_Color_Recognition_digital	Steuerung mit Farbflächen (A8/A4)
Buggy_Line_Follower_with_Obstacle_Detection_digital	Spurfolger mit Hinderniserkennung (A8/A3)
Buggy_Line_Follower_with_P_Controller	Spurfolger mit Proportionalregler (A8/E2)
Buggy_Line_Follower_with_PD_Controller	Spurfolger mit PD-Regler (A8/E3)
Buggy_Line_Recognition	Spurenenerkennung mit Kamera (A8/E1)
HEX_Colorcode	Farbwert als HEX-Code ausgeben

6. Autonomous Driving

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/559896-robotics-add-on-autonomous-driving>



Autonomous_Parking	Automatisches Einparken (A4/E1)
Display_Counter_Encoder	Impulszähler Encodermotor (A1/A1)
Distance_Test	Distanzzähler Encodermotor (A1/A2)
Odometer	Kilometerzähler (A1/A2c)
Tachometer_and_Odometer	Tachometer und Kilometerzähler (A1/A3)
Tachometer_and_Odometer_Thread	Tachometer und Kilometerzähler im Thread ausführen (A1/E1)
Taxi_Meter	Taxameter (A1/E2)
Brake_Assist	Bremsassistent (A2/A3)
Emergency_Stop	Not-Halt (A2/A1)
Lane_Keeping_Assist_P_Control	Spurhalteassistent (A2/E1)
Lane_Keeping_Assist_PD_Control	Spurhalteassistent (A2/E2)
Servo_Calibration	Geradeausfahrt (A2/A2)
Speed_Control	Tempomat (A2/A4)
Automatic_Lighting	Automatische Lichtfunktionen in Threds
Blinker	Blinkertest (A3/A3)
Blinker_with_Warning_Lights	Blinkertest mit Warnlicht (A3/E1)
Headlight	Abblendlicht (A3/E2)
Reversing_Light	Rückfahrlicht (A3/A1)
Stoptlight	Bremslicht (A3/A2)
Stoptlight_hangover	Bremslicht mit Nachleuchten (A3/A2b)
Stoptlight_proportional	Bremslicht mit proportionaler Helligkeit (A3/A2c)
Find_Parking_Space	Parklücke finden (A4/A1)
Parking_Manoeuvre	Einparkmanöver (A4/A2)
Tachometer	Tachometer (A1/A3)

7. Competition

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/560842-robotics-add-on-competition>



gui_sensor_gesture	Python-Code	Test Gestenmodul
gui_sensor_combi		Test Kombisensor

8.IoT

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/559897-robotics-add-on-iot>



Sensorstation	Sensordaten an Cloud übermitteln	
IoT_Barometer	Anzeige Luftdruck, Luftfeuchtigkeit, Temperatur (A1/A2)	
IoT_Barometer_with_NTC_resistor	Anzeige Luftdruck, Luftfeuchtigkeit, Temperatur mit Sensor und NTC (A1/E1)	
IoT_Metrological_Station	Metrologische_Station (A1/A1)	
IoT_MQTT Python-Code	Metrologische_Station über MQTT (Temperatur, Luftfeuchte und Druck)	
IoT_MQTT_Barometer Python-Code	Metrologische_Station über MQTT (Temperatur, Luftfeuchte und Druck) (A1/E2)	
IoT_MQTT_Barometer_with_Webcam Python-Code	Metrologische_Station über MQTT inkl Webcam (A1/E3)	
IoT_Webcam Python-Code	Bild mit Kamera aufnehmen	
IoT_CO2_Signal_Light	Lufqualitätsmessung (CO ₂ Ampel) (A2/E2)	
IoT_Illuminance	Anzeige Beleuchtungsstärke (A2/E1)	
IoT_Indoor_Air_Quality	Anzeige Luftqualität (A2/A2)	
IoT_Indoor_Climate	Anzeige Temperatur und Luftfeuchtigkeit (A2/A1)	
IoT_Loudness	Anzeige der Lautstärke auf GUI und Kontrolllampen (A2/E3)	
IoT_MQTT_Indoor_Air_Quality Python-Code	Anzeige Luftqualität	
IoT_Camera_Motion_Control	Grundstellung Kamera	
IoT_Init_Camera_A	Grundstellung Variante A (A3/A1c)	
IoT_Init_Camera_B	Grundstellung Variante B (A3/A1c)	
IoT_Init_Camera_Pan	Grundstellung Variante C (A3/A1c)	
IoT_Init_Camera_Tilt	Grundstellung Kamera (A3/A1)	
IoT_Microphone_Level	Geräuschaktivierung (A3/A3)	
IoT_Surveillance_Camera Python-Code	Kameraüberwachung (A3/A2)	
IoT_Surveillance_Camera_Dashboard_Control Python-Code	Steuerung über das Cloud-Dashboard (A3/E2)	
IoT_Surveillance_Camera_Dashboard_Control_Test Python-Code	Kameraüberwachung mit Fernsteuerung Test	
IoT_Surveillance_Camera_Motion_Detection Python-Code	Bewegungserkennung (A3/A4)	
IoT_Surveillance_Camera_Noise_Detection Python-Code	Geräuchererkennung mit Schwellwert (A3/A3)	
IoT_Surveillance_Camera_Voice_Control Python-Code	Sprachsteuerung (A3/A1)	
IoT_Test_Dashboard_Control Python-Code	Test zwischen IoT und Dashboard	

9.Omniwheels

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/559898-robotics-add-on-omniwheels>



Mecanum_Boundary_Line	Spurfolger (Linienerkennung) (A1/A4)
Mecanum_Line_Follower_digital	Spurfolger mit Spursensor (A2/A1)
Mecanum_Synchronous_Navigation_Distance	Synchroner Antrieb mit Distanz-Vorgabe (A1/A3)
Mecanum_Synchronous_Turning_Functions	Synchrones Drehen (A1/A2)
Mecanum_Synchronous_Driving_Functions	Synchroner Antrieb in alle Richtungen (A1/A1)
Mecanum_Collision_Prevention	Hinderniserkennung mit Ultraschall (A1/E1)
Mecanum_Line_Follower_with_Color_Recognition_digital	Spurfolger mit Farbsteuerung (A2/A3)
Mecanum_Line_Follower_with_PD_Controller	Spurfolger mit PD-Regler (A2/E2)
Mecanum_Target_Reacher	Encoder-Navigation (A1/E2)
Mecanum_Line_Follower_Obstacle_digital	Spurfolger mit Hinderniserkennung (A2/A2)
Mecanum_Line_Follower_P_Controller	Spurfolger mit Proportionalregler (A2/E1)
Mecanum_Coordinates_Drawing	Malen nach Zahlen (A3/E2)
Mecanum_Drawing_Coordinates	Zeichnungskordinaten
Mecanum_Drawing_Polygons	Zeichnen eines N-Ecks (A3/A3)
Mecanum_House_of_Santa_Claus	Zeichnen Haus vom Nikolaus (A3/A2)
Mecanum_Move_2_Point	Zielpunkt ansteuern (A3/E1)
Mecanum_Correct_Distance	Zielabstandskorrektur (A4/E2)
Mecanum_Find_Target	Zielsuche (A4/E4)
Mecanum_Fire_and_Load	Auswurfmechanismus (A4/A1)
Mecanum_Fire_and_Load_extended	Auswurfmechanismus mit Füllstandsanzeige (A4/A1)
Mecanum_Fire_and_Load_Voice_Control	Sprachsteuerung (A4/A2)
Mecanum_Turn2Target	Zielausrichtung (A4/E3)
Test_Target_Distance	Abstandsmessung (A4/E1b)

10. Künstliche Intelligenz

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/569022-add-on-kuenstliche-intelligenz>



Add_On_AI Python-Code	Sortierstrecke KI
TM_A1	Grundprogramm
TM_A2 Python-Code	Bilder_sammeln
TM_A3 Python-Code	Bilder_sammeln
TM_A4 Python-Code	Bilder_erkennen
TM_A6 Python-Code	KI_sicher
A1_Barrier_core_programm Python-Code	Schranke_Grundprogramm_AI
A2_Barrier_train Python-Code	Schranke_trainieren_AI
A3_Barrier_control_AI Python-Code	Schranke_steuern_AI

11. Industrial Robots

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/564064-add-on-industrial-robots>



Industrial_robots_teachin Python-Code	Noch nicht Dokumentiert
Industrial_robots_2a	Referenzfahrt einer einzelnen Achse
Industrial_robots_2b	Referenzfahrt mit Servomotor
Industrial_robots_2c	Referenzfahrt aller Achsen
Industrial_robots_3a	Ansteuerung der Encodermotoren
Industrial_robots_3b	Umrechnung von Grad in Impulse
Industrial_robots_3c	Kombination der Unterprogramme
Industrial_robots_4b	Überprüfung der Rechnung
Industrial_robots_6b Python-Code	Pick and Place mit Punktsteuerung
Industrial_robots_8b Python-Code	Greifaufgabe

12. Coding Competition

<https://www.fischertechnik.de/de-de/produkte/schulen/mint-robotics/571099-stem-coding-competition>

Dieser Bausatz ist speziell für Robotik-Wettbewerbe konzipiert worden und enthält keine Unterrichtsmaterialien oder Lehrerunterstützung.



13. RX-Controller

Der Controller ist z.Z. nur im Komplettpaket als Smart Robots Max zu erwerben. (Siehe 14)



RX_test_gesture_sensor	Testprogramm für Gestensensor (nur die Gesten und Entfernung))
test_RX_i2c_device_apds9960 Python-Code	Testprogramm für Gestensensor (Farbwerte)
test_RX_i2c_device_scan Python-Code	Erkennung der I ² C-Adresse
test_RX_i2c_device_template Python-Code	Grundlage für eigene I2C Sensoren

14. Smart Robots Max

<https://www.fischertechnik.de/de-de/produkte/spielzeug/robotics/564111-smart-robots-max>



2-wheeled_robot_1 - 2	2-Radroboter 1 - 2
Forklift_1 - 2	Gabelstapler 1 - 2
Advanced_2-wheeled_robot_1 - 4	Erweiterte_2_Radroboter 1 - 4
Ball_robot_1 - 3	Ballroboter 1 – 3
Twilight_switch_1 - 3	Dämmerungsschalter 1 – 3
Optical_illusion_1 - 2	Optische Täuschung 1- 2
Omniwheels_Base_Model_1 - 3	Omniwheels Basismodell 1 – 3
Omniwheels_track_gesture_sensor_1 - 4	Omniwheels mit Spur- und Gestensensor 1 - 4
Dancing_robot_1 - 2	Tanzroboter 1 - 2

15. Smart Robots Pro

<https://www.fischertechnik.de/de-de/produkte/spielzeug/robotics/569021-smart-robots-pro>



BTSmart_Robot_TeachIn	Lernender Roboter
BTSmart_Carousel_1 - 4	Karussell 1 - 5
BTSmart_Useless_Machine_1 - 2	Nutzlose Maschine 1 - 2
BTSmart_Dance_robot_1 - 2	Tanzroboter 1 - 2
BTSmart_Drivebot_1 - 2	Automatischer Fahrroboter 1 - 2
BTSmart_Alarm	Alarmanlage
BTSmart_Fan_1 - 2	Ventilator 1 - 2
BTSmart_Spare_parts_dispenser_1 - 2	Bauteilspender 1 - 2
BTSmart_Ball_game_1 - 2	Ballspiel 1 - 2
BTSmart_Boat_swing_1 - 2	Schiffschaukel 1 - 2

Die **Aufgaben** sind im Dokument Challenge unter <https://www.fischertechnik.de/-/media/fischertechnik/rebrush/spielzeug/e-learning/smart-robots-pro/challenges-smart-robots-pro-de.pdf> Beschrieben.

Die **Bauanleitungen** sind als Animationen unter <https://www.fischertechnik.de/de-de/spielzeug/e-learning/smart-robots-pro> zu finden.

16.BT-Smart Beginner Set

<https://www.fischertechnik.de/de-de/produkte/spielzeug/robotics/540586-bt-smart-beginner-set>



BTSmartBeginnerSet_barrier	Schranke
BTSmartBeginnerSet_conveyor_belt_1 - 3	Förderband 1 - 3
BTSmartBeginnerSet_conveyor_belt_punch_1 - 2	Förderband mit Stanze 1 - 2
BTSmartBeginnerSet_hand_dryer	Händetrockner
BTSmartBeginnerSet_hindrance_detector	Hinderniserkenner
BTSmartBeginnerSet_lighthouse_1 - 3	Leuchtturm 1 - 3
BTSmartBeginnerSet_merry_go_round_1 - 3	Karussell 1 - 3
BTSmartBeginnerSet_mobile_robot_1 - 4	Fahrroboter 1 - 4
BTSmartBeginnerSet_pedestrian_light_1 - 2	Fußgängerampel 1 - 2
BTSmartBeginnerSet_punch_with_safety_switch	Stanze mit Sicherheitsschalter
BTSmartBeginnerSet_refrigerator_1 - 2	Kühlschrank 1 - 2
BTSmartBeginnerSet_washing_machine_1 - 5	Waschmaschine 1 - 5

Die **Bauanleitung** liegt unter:

https://fipproductmedia.azureedge.net/media/Marketing%20Materials/Operating%20instructions/BA_540586_BT_S MART_BEGINNER.pdf

17.STEM Coding RoboMission

<https://www.fischertechnik.de/de-de/produkte/schulen/robotik/576109-stem-coding-robomission>



RoboMission_linepid Python-Code	Linienfolger mit PID-Regler
RoboMission_display_start	Beispiel für grafische Oberfläche (GUI)
RoboMission_motor_servo	Beispiel für Motoren- und Servosteuerung
RoboMission_line	Beispiel für Linienverfolgung mit Farbsensoren
RoboMission_rgbw_hsv Python-Code	Beispiel für das Auslesen des RGB-Farbsensors
RoboMission_template Python-Code	Vorlage für eigene Projekte

Die **Bauanleitung** liegt unter:

https://fipproductmedia.azureedge.net/media/Marketing%20Materials/Operating%20instructions/BA_576109_BAUANLEITUNG-STEM-CODING-ROBOMISSION.pdf