FTmodule Examples

Author: Jeroen Regtien
Version: V1.0
December 2025

The FTmodule library is an extension of the FTlegacy library allows selected Arduino based microcontrollers to control one or more module fischertechnik interfaces or third part, such as the parallel (universal) or serial(intelligent) interfaces. It is also capable of using Arduino based Controllino PLCs as well as controlling the Didacta shields (Mega and Uno), the Adafruit motorshield and the ftNano shield. It will run fine on Arduino Mega's en Uno's R4, but probably not always on the Uno R3. A special case is the ftDuino, this is already a combination of an Arduino and an Input/Output shield totally designed for fischertechnik and a great solution for those who want to work in the Arduino environment with fischertechnik. The ftDuino has its own 'ecosystem' with excellent documentation and examples and was included in the scope of this project for the sake of portability of my software. The ftDuino can do much more though than the functionality offered in FTmodule. A description of how to connect these interfaces to an Arduino is provided in the appendices of the software manual.

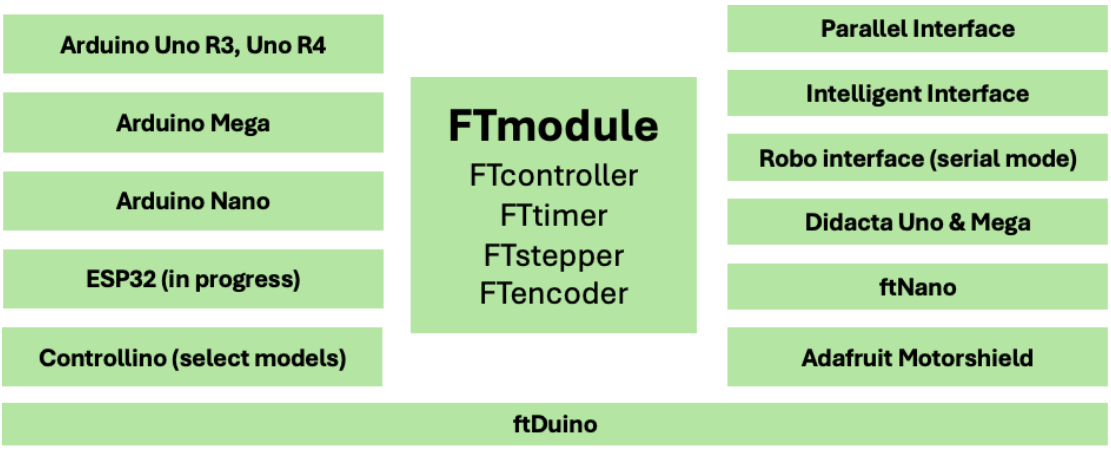
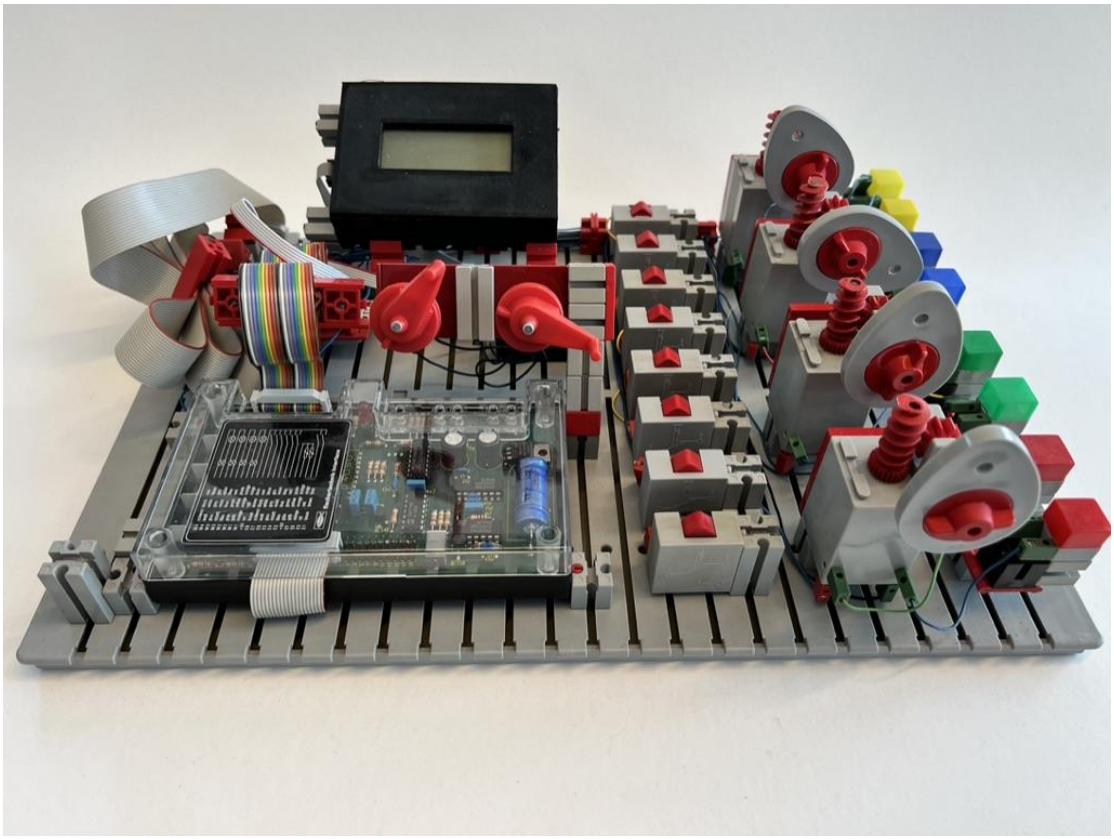


Table of Contents

Basic Tests

The basic tests for both a serial as a parallel interface are done with a setup consisting of a base plate with four motors, 8 lights, 8 input switches and two potentiometers as can be seen in the



The lights are connected with one side to a motor output, the other one to GND. In this way a different light will light up when a motor changes direction. Based on the input given a motor/light is activated.

Function:

EX>100	O1 = on, else O1 = off
EY> 100	O2 = on, else O2 = off
E3	O3 = on, O4 = off
E4	O4 = on, O3 = off
E5	M3 CW
E6	M3 stop
E7	M4 CW
E7+E8	M4 CCW
E8	all outputs and motors off

Programs:

FT_basic_par	for parallel interface
FT_basic_ser	for serial interface
FTmodule_ftDuino_test	A simple ftDuino test program

Computing (30554)

The first computing kit with some 9 models introduced by fischertechnik in 1984. For a description of the kit see:

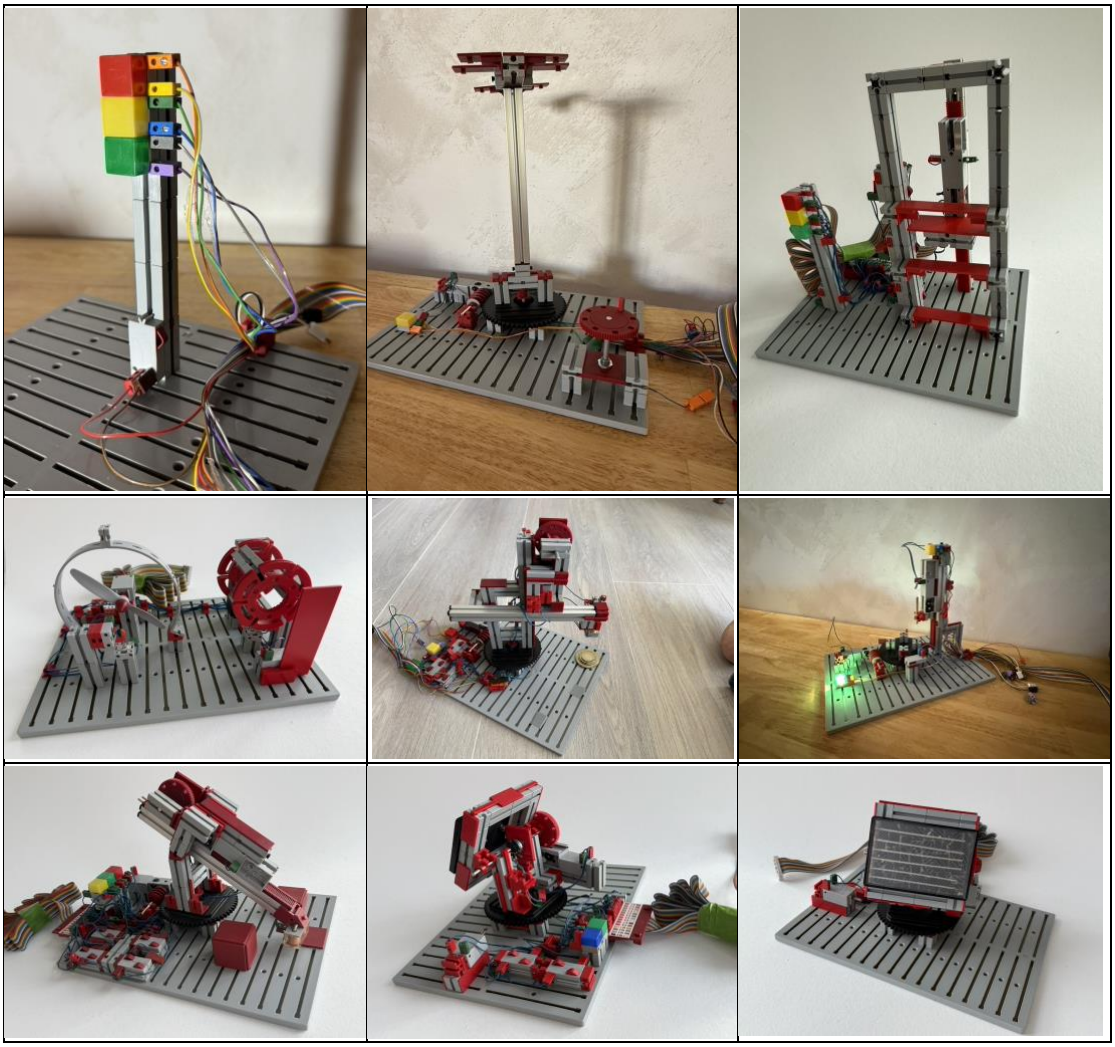


Figure 2. Computing models

Programs (for inout/output connections check example program):

FT_computing_Ampel	Operates the traffic light using one button and three lights. Uses FTtimer objects.
FT_computing_AntennenRotor	Moves the antenna based on the setting of the other potentiometer. Uses two analog input signals.
FT_computing_Lift	Three floor lift model
FT_computing_SunPosition	Elaborate model that tracks the sun position. It requires a connected LCD display and uses the four switches to obtain the geographical position, date/time etc to calculate the correct azimuth and elevation. It then tracks the position of the sun from that moment onwards. Reference information on how the algorithm works is enclosed in the example directory.

FT_computing_TowerHanoi	The tower of Hanoi. This model probably requires some adjustments and use of tape to get the magnet to release properly. Recursion however is such a beautiful programming technique where with a minimum of code complicated repetitive movements can be executed.
FT_computing_Ventilator	A simple ventilator that kicks in when the NTC, heated by the lenslamp, triggers threshold for the analog input signal.
FT_computing_Washmachine	A simulated washing machine. When the door closes, the water is heated. When the right temperature is reached (analog NTC signal) the back-and-forth rotations begin. Rotations are counted with a switch.
FT_computing_Werkzeug	A simulated drill operating on a rotating platform
FT_computing_Teachin	The most interesting model from this kit. With the 8 input switches the model can be programmed: E1: execute demo sequence E2: execute learned sequence E3: rotation counterclockwise E4: rotation clockwise E5: arm up E6: arm down E7: toggle magnet on/off E8: learn current position After building an connecting the model, fisrt a calibration has to be performed to determine the best location settings.
FT_Antennenrotor_Double	Two antenna rotors each connected to an interface, in turn connected to a single Arduino Mega with a Megashield.

Plotter (30571)

The novelty of the box, which appeared in 1985, consisted of the two step-motors that could be driven by a parallel interface. More information on how to use these stepper motors in combination with fischertechnik is described in:

fischertechnik: Hinweise zum fischertechnik-schrittmotor 32311.
https://docs.fischertechnikclub.nl/computing/79015.pdf

fischertechnik – InstructionsPlotter/Scanner 39460 or 39462
https://docs.fischertechnikclub.nl/computing/39460a.pdf
https://docs.fischertechnikclub.nl/computing/39462.pdf

The FTlegacy and FTmodule libraries contain an FTstepper and FTstepperXY class, where the latter can be used to drive a plotter, 2D scanner or CNC equipment.

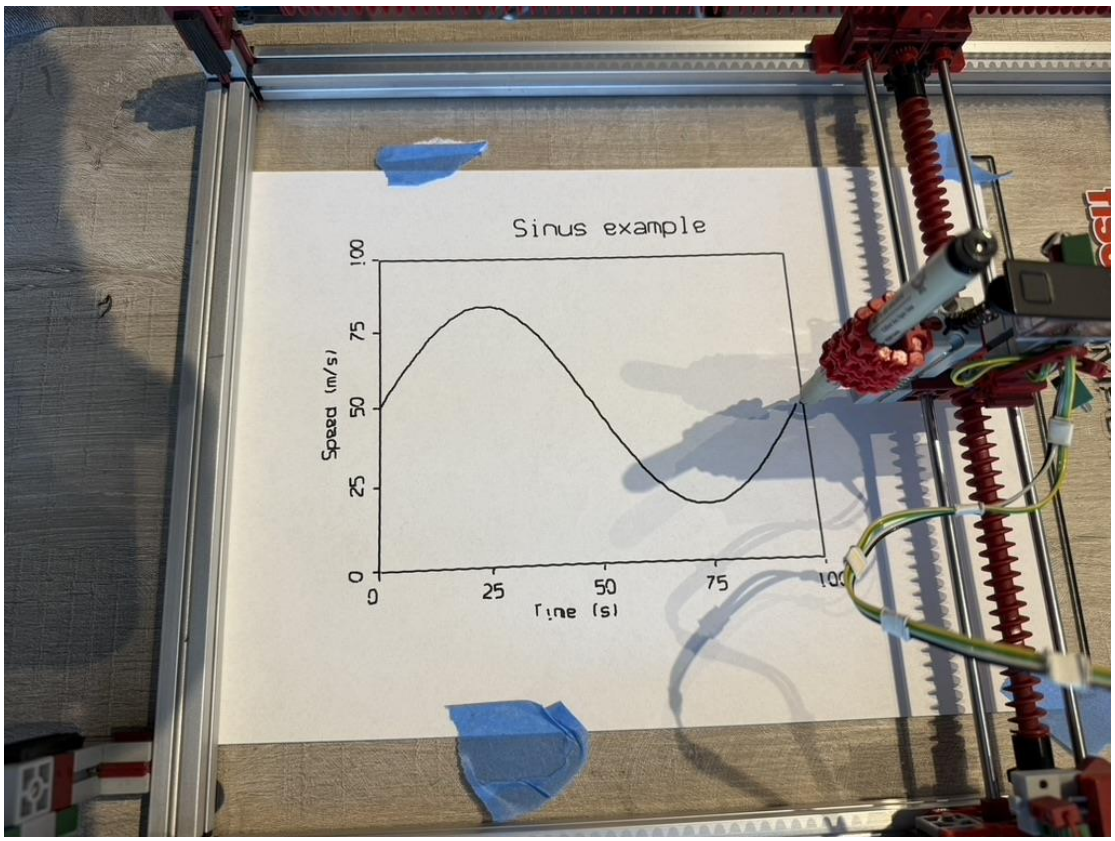
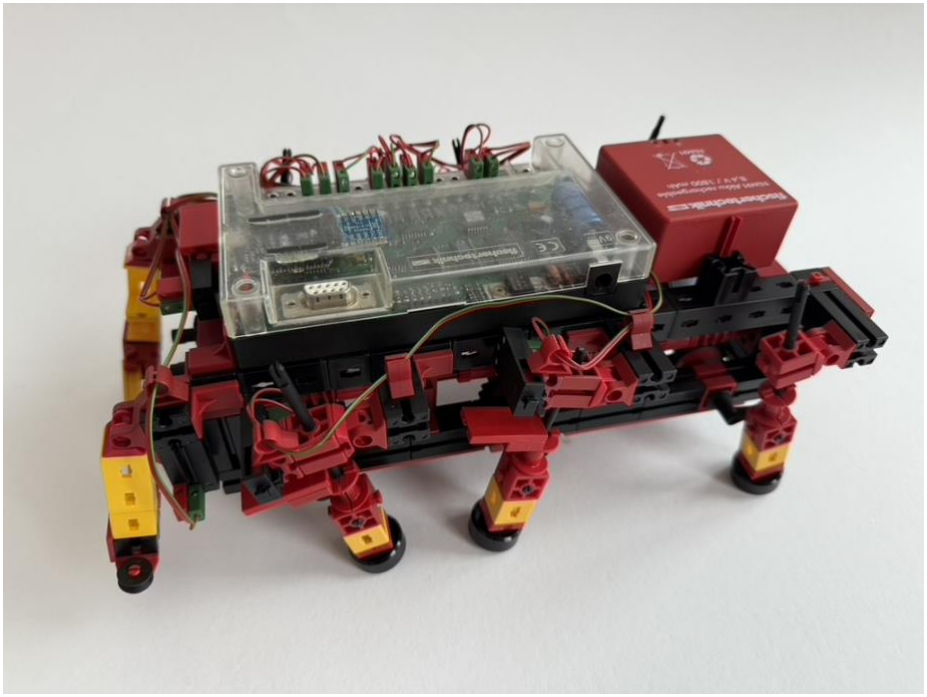


Figure 2. Result of 'Plotter_sinus'

Plotter_sinus	Draw XY plot with sinus curve. The routine uses a box, axis, plotting of text and plotting a curve
Plotter_xmas	Draw x-mas tree with New Year's wish. An array with the coordinates of the outline of the x-mas tree is defined as well as the position of the balls in the tree. Finally text strings are plotted.

Bionic (57486)

The Computing Bionic Robot Computing kit (57486) from 2001 contains a number of models. Further information can be found in:
https://docs.fischertechnikclub.nl/digital/62964.pdf
https://docs.fischertechnikclub.nl/digital/62965D.pdf



The example taken is “Jack”. The interesting part of these models is that without synchronization the model would not be able to walk straight or topple over because will motors run at slightly different speeds for a variety of reasons. By checking the movement of the legs with input switches both sides of the robot will move in a synchronized fashion.

In the test model a serial interface with an embedded Arduino Mini is used. The robot is controlled through commands on the serial monitor. Als check the connections and labelling within the program; a slightly different choice has been made for the labelling of the input switches.

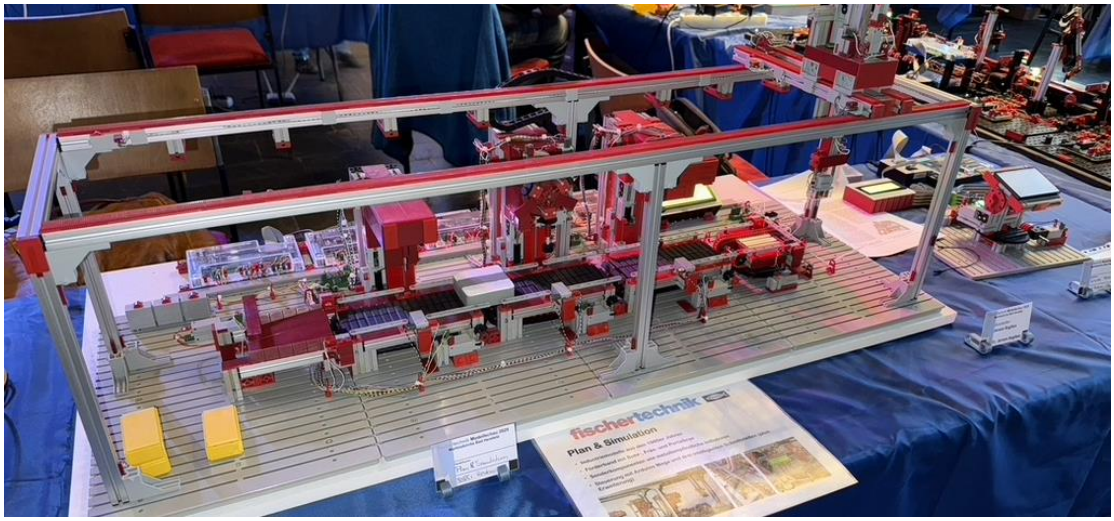
Looking from above with the battery backwards, right side:
Motor: ft_M1 ; Front leg switch: ft-E1 ; Middle leg switch: ft-E3 ; Bumper: ft-E5
Left side: Motor: ft_M2 ; Front leg switch: ft-E2 ; Middle leg switch: ft-E4 ; Bumper: ft-E6

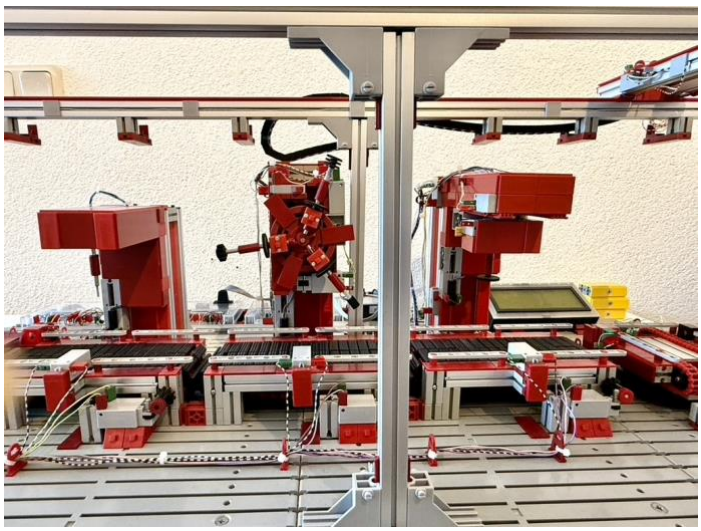
Bionic_Jack	The robot is operated by the following serial monitor commands: ‘f’ : forward, ‘b’ : backward, ‘r’ : right, ‘l’-left, followed by a single digit to indicate the number of steps. The Bumper switches are not yet operated upon. The idea is there to let Jack walk until he encounters an obstruction to then take evasive action, a nice programming puzzle.

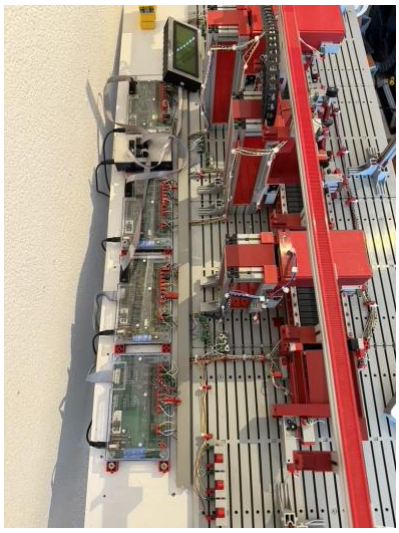
Workstations

Plan & Simulation Taktstrasse

This model was sold by fischertechnik as a Plan & Simulation model number 30581. It has been extensively described in an ft:pedia article:
https://www.ftcommunity.de/ftpedia/2025/2025-3/ftpedia-2025-3.pdf#page=44







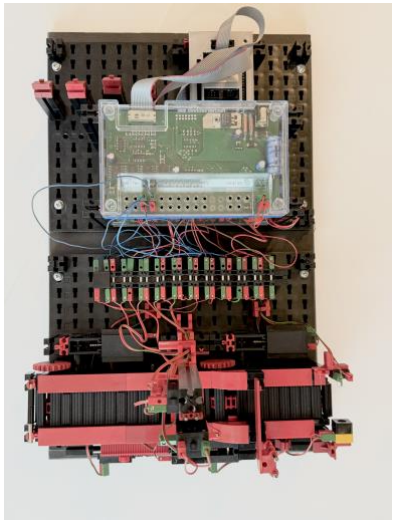
The control is via three serial interfaces, of which one has an extension unit. The controller is an Arduino Mega with a connected 2004 LCD. The model contains 20 bi-directional motors, 9 uni-directional outputs, 20 switches, 3 initiators and 2 photocells.

The software loops continuously over all the tasks and keeps track of which tasks are completed. Only then it starts the next task. There is an interrupt button such that the sequence of events can be stopped at any time.

FT_WKS_30581_total.ino	The full program

Rotating belt with drill

This model is part of an educational project that develops modular units that can be connected to each other. The educational task is to program the controller and make the connections from the terminal at the side of the model to the controller/interface. These models can be equipped with a variety of controllers and shields. There are movr examples in set of FTmodule examples.



A work piece enters the belt on the left via a photocell, the belt transports the piece to a resting plate and the drill comes down to drill a hole. Then a lever shifts the piece onto the second belt and the piece move to the next station.

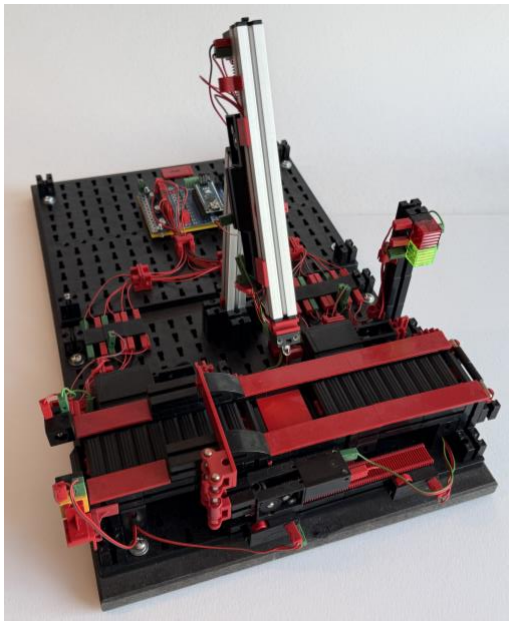
In this model a serial interface with an extension unit is used. The software loops continuously over all the tasks and keeps track of which tasks are completed. Only then it starts the next task.

FT_WKS_Drill_SEREX.ino	The full program

Rotating belt with welding arm

In the same modular series a workstation was created with a welding station. The piece is trnasported once it is recognised by the lightcell t the beginning of the belt and dropped in the middle. The the arm lowers and a simulated welding action is executed. Afterwards the arm is raised and the lever pushes the piece on the rotating belt on the right. Two lights (green, arm) indicate status of the operation.

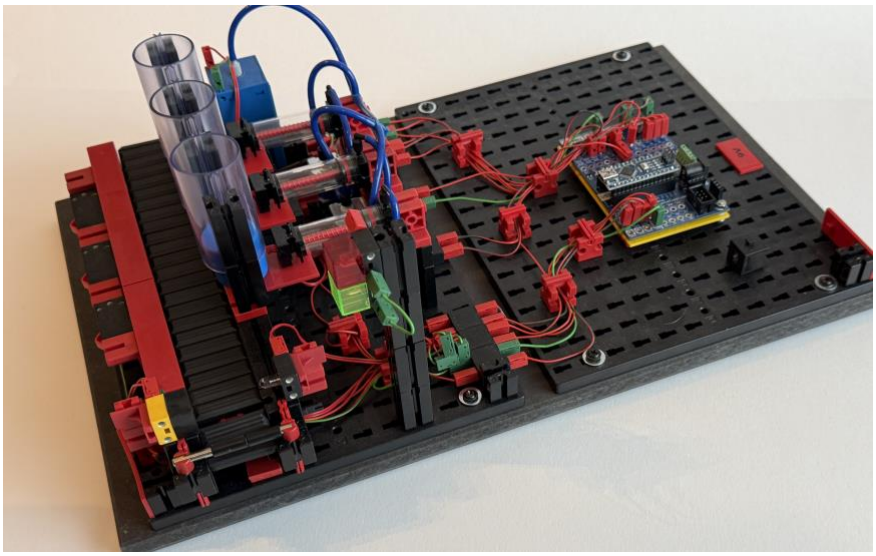
This particular example uses a ftNano, there is also a slightly different ftDuino version

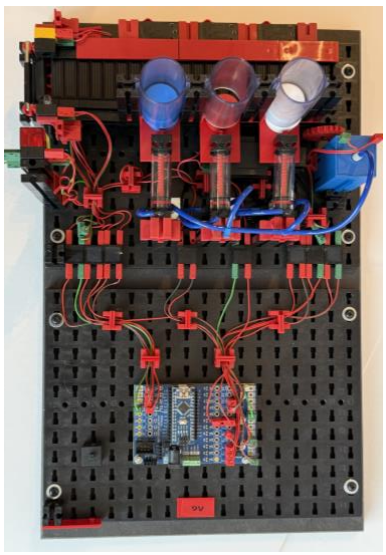


FTnano_welding	The full program for the ftNano
FTmodule_ftDuino_weld	A slightly different version for the ftDuino

Rotating belt storage release unit

In the second ftNano example and in the same modular series a workstation was created that froms the beginning of the series. Upon pressing one of three switches a piece is pushed by a pneumatic piston on the belt, which moves to the right. At the end the next workstation picks up the piece and the lightcell triggers a stop of the belt after the piece has passed.

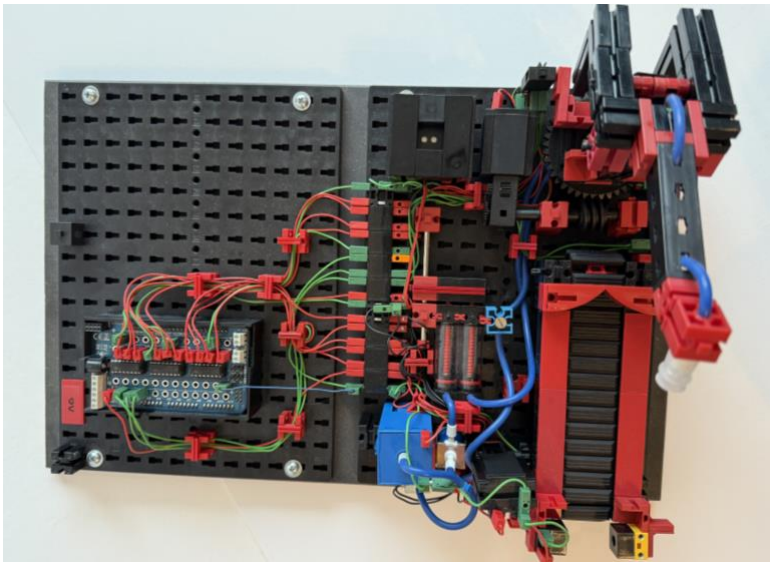




FTnano_storage	The full program

Rotating belt color identification

This workstation is based on the color identification model of the TXT ElectroPneumatic kit (121936). The piece is transported to a pickup point, where the pneumatically controlled rotating arm picks up the piece and transports it to the measurement box. Upon determining the colour the piece is rotated further to the next workstation. The color of the workpiece is indicated by one of the three lights.



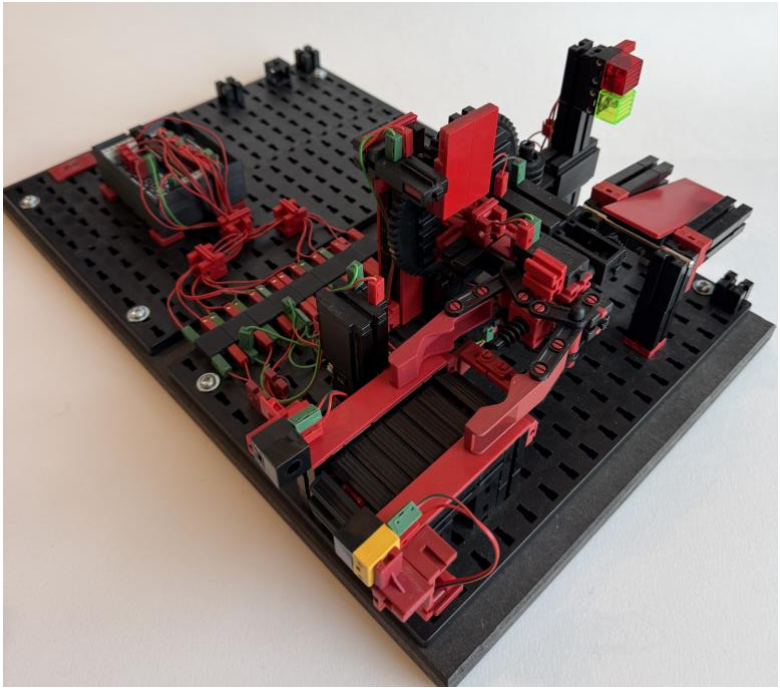
As a controller / shield the Arduino Mega and the Didacta Mega shield are used.

FT_WKS_Color_DID_MEGA	The full program

Rotating belt turning station

This workstion picks up a piece, flips it and releases in on a slide to the next workstation.

It uses a Arduino Uno and Didacta Uno combination.

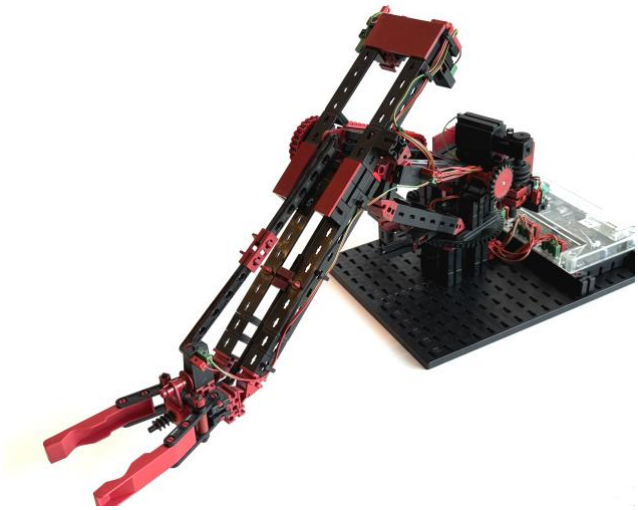


FT_WKS_Turn_DID_UNO	The full program

Robot Arms

Industry Robot Profi Computing (30490)

This robot arm comes from the Profi Computing kit issued in 1991. Is was accompanied by a fairly sizeable book “Experimentierbuch Profi Computing” which should be referenced: https://docs.fischertechnikclub.nl/digital/36069.pdf and is also available in English (https://docs.fischertechnikclub.nl/digital/36072UK.pdf) or Dutch (https://docs.fischertechnikclub.nl/digital/36072NL.pdf)



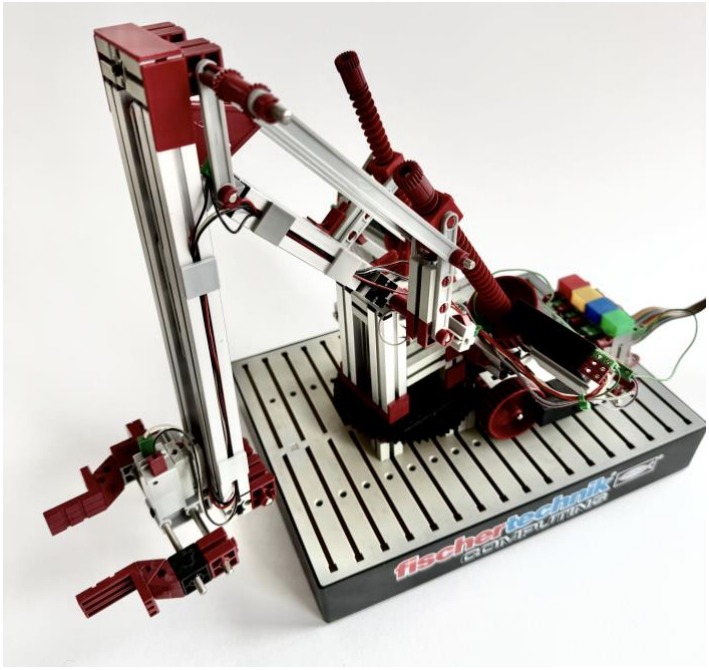
This robot arm only has two degrees of freedom, it can rotate and the arm can move up or down. Because of a lever construction the gripper changes direction when the arm is lowered.

FT_30490_arm	A demonstration program that uses the encoder class for the three motors and takes commands from the serial monitor. The commands ‘d’, ‘u’, ‘l’, ‘r’ can be followed by a 2 digit number to indicate the number of steps. 'd' : arm down 'u' : arm up f5 means five steps forward. 'l' : rotate left 'r' : rotate right 'c' : close gripper 'o' : open gripper 'i' : reset robot arm 'm' : perform a movement of two stapled bricks 's' : all motors stop The command ‘m’ starts a small demo routine where to stapled brick are moved.

Training Robot (30572)

This is one of the iconic fischertechnik computing models issued in 1987. It uses rotating discs with black stripes that rotate through a photocell and thus create pulses. The pulses that are then counted so that desired positions can be realized. The arms have switches that determine the home position from which the counting takes place.

More information on this model can be found in:
https://docs.fischertechnikclub.nl/computing/39461.pdf (German)
https://docs.fischertechnikclub.nl/computing/39461.pdf (English)
https://docs.fischertechnikclub.nl/computing/32368.pdf
https://docs.fischertechnikclub.nl/info2/30572.pdf



FT_30572_basic_encoder	A test program is made based on the FTencoder class that controls the rotation and the two arm motors. Via the Serial Monitor a number of commands can be given, such as: 'd' : lower arm 'u' : raise arm 'f' : arm forward 'b' : arm backward 'l' : left rotation 'r' : right rotation 'o' : gripper open 'c' : gripper close 'i' : reset all. 'm' : execute demo Note that the end-switches for the rotation and arm have been altered, they are now normally open (NO) instead of normally closed (NC). After the 'd,u,f,b,l,r' commands a number <99 can be given for the number of steps to be taken. The program contains the routines to check illegal

	combinations of the upper and lower arms.
FT_30572_hanoi	This program is the same as the above text program with added functions to execute the Tower of Hanoi puzzle which can be invoked by the 'h' command in the serial monitor. Unfortunately, there is quite some hysteresis which undermines the accuracy. If one moves the arm back and forth a few times, the rotation position starts to deviate. There is not yet a software solution for this.

Note:

- When using something different than a parallel interface, make sure the power to the counters light cells does not exceed 5V. See FT Community forum for details
- Make sure there is a ground cable between the interface and the light cells (Gabellichtschranke)

Industry Robots II (96782)

This model comes from the kit Industry Robots II. It uses a power motor, two S-motors and and an XS motor for the gripper. More information can be found in: https://docs.fischertechnikclub.nl/digital/113917.pdf, and https://docs.fischertechnikclub.nl/digital/114306D.pdf

FT_96782_arm_test	This is kind of a teach-in robot. The software assumes the presence of three platforms, the size of a fischertechnik cassette. The location relative to the robot arm can be seen in the photo. Via the serial monitor interface commands can be given to make any movement. ‘f’ : forward, ‘b’ : backward, ‘r’ : right, ‘l’: left, ‘l’ : left and ‘r’ : right followed by a double-digit number to indicate the number of steps. In addition: ‘c’: close gripper ‘i’ : initialise robot ‘m’ : execute demo program. Place a yellow cylinder on the middle platform.
FT_96782_hanoi	This program builds on the abovementioned arm_test program and in addition has functions to solve the Tower of Hanoi puzzle. The puzzle can be solved by pressing the letter ‘h’ in the serial monitor.

Fox/Puttmann Robot Arm

This is the robot arm as described in the book fischertechnik-Roboter mit Arduino by Dirk Fox and Thomas Püttmann (dpunkt Verlag, 2020).

The interesting thing here is that the location of the arms is measured through three potentiometers which give an analog signal. This resembles the method used in the Teach-In robot of the fischertechnik Computing kit(30554)

FT_Adafruit_RobotArm	A number of commands can be given through the Serial Monitor: 's' : all motors stop ‘i’ : initial positions / reset 'd' : lower arm (elbow) down 25 steps ‘D’ : lower arm to positon 300 'u' : lower arm (elbow) up 25 steps ‘U’ : lower arm to positon 100 'f' : upper arm (shoulder) forward 25 steps

	‘F’ : upper arm to position 500 'b' : upper arm (shoulder) backward 25 steps ‘B’ : upper arm to position 150 'l' : rotate left 25 steps ‘L’ : rotate left to position 300 'r' : rotate right 25 steps ‘R’ : rotate right to position 400 'c' : close gripper 'o' : open gripper ‘m’ : excute demo The demo moves a yellow cylinder back and forth. The starting position can bee determined by running the program a first time.
FT_Adafruit_RobotArm_Position	For various movements it is necessary to determine the right coordinate settings. This simple program helps. First remove the small cogwheel so that the arm can move freely. Upon pressing a switch connected to input one (ft_E1) the coordinates are printed in the serial monitor. This program is essential for getting the Tower of Hanoi code right
FT_Adafruit_RobotArm_Hanoi	The same program as the abovementioned RobotArm with the addition of the logic and recursion to solve the Tower of Hanoi puzzle. All the logic is there, however the coordinates for each of the piles must be added. These depend on the size of the discs and the number of discs. At the top of the program the constant ‘numDiscs’ has to be set followed by the series of coordinates for each position. This is quite a puzzle and the ‘Position’ program above has to be used.