

ROBOTİK LABORATUVARI EKLENTİ KİTAPLARI

LİSE
LE2

Programlama Dilleri:

C, C++

python

OPEN
ROBERTA

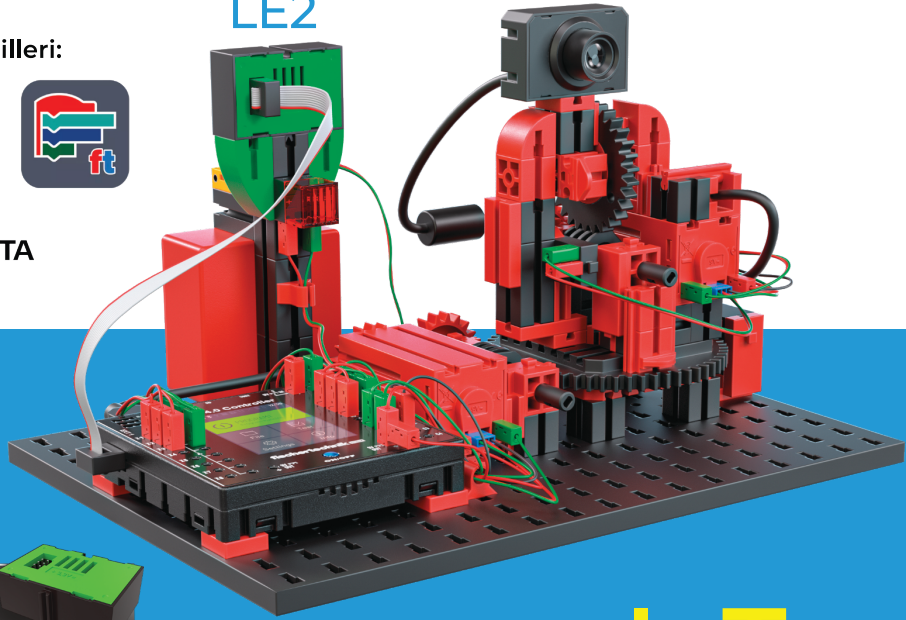
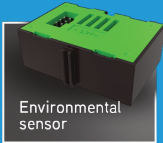


Photo resistor



CLOUD



Environmental
sensor

IoT

NESNELERİN İNTERNETİ



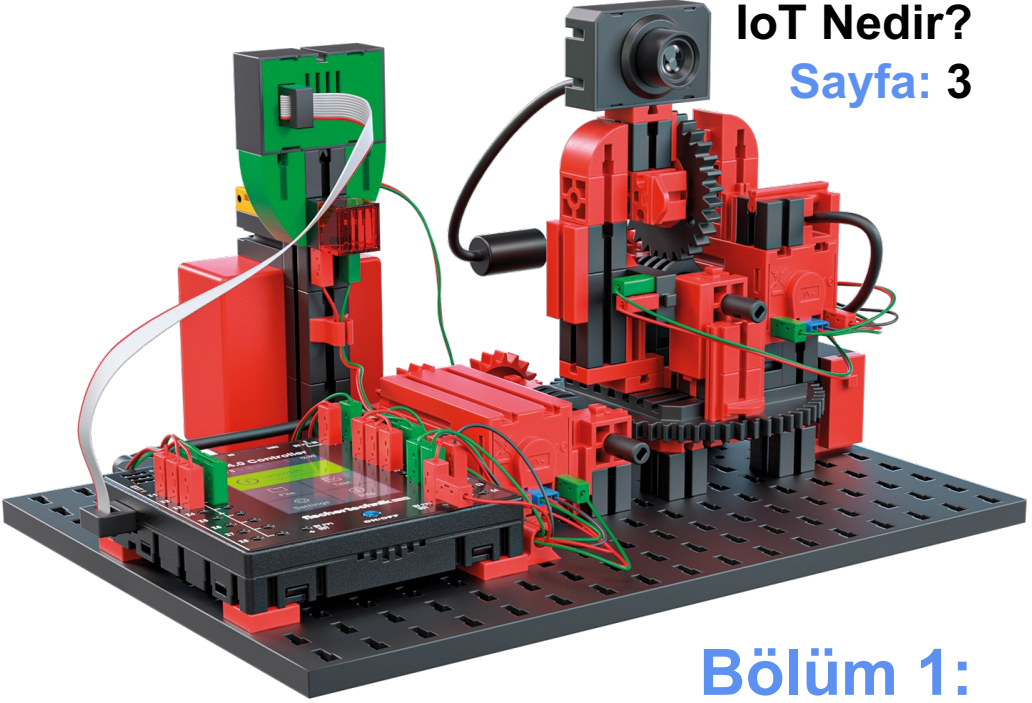
STEM **M**ATHEMATİK
INFORMATİK
NATURWISSENSCHAFT
TECHNIK

SCIENCE TECHNOLOGY ENGINEERING MATHEMATICS

GİRİŞ:

IoT Nedir?

Sayfa: 3



Bölüm 1:

IoT Modeli

Sayfa: 6

- **Hava Durumu İstasyonu**

Sayfa: 19

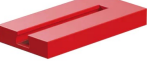
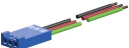
- **İç Mekan İklimi**

Sayfa: 30

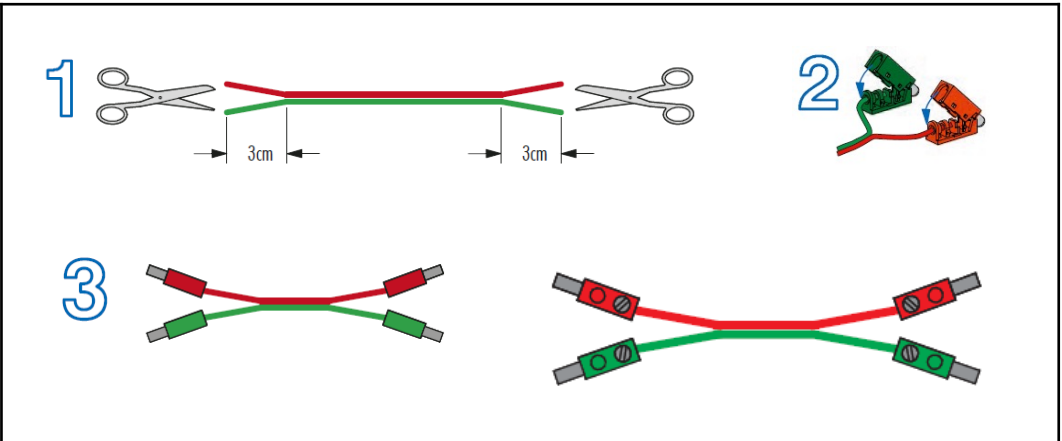
- **Alarm Sistemi**

Sayfa: 43

Parça Listesi

 15	31 060 3x		32 881 6x		32 698 1x		35 072 1x
	31 390 1x		35 049 1x		31 771 1x		36 264 1x
	31 391 1x	 30	35 063 3x		37 468 6x	 60	38 416 1x
	31 602 2x		163 201 2x		137 677 4x		181 583 3x
	32 330 2x		163 202 2x		36 334 2x		181 584 4x
	32 064 3x		182 974 1x		38 216 1x		135 465 1x
	31 982 11x		168 961 1x				

Kablo Bağlantıları



GİRİŞ

IoT Nedir?



Öğrenme hedefi

- ✓ IoT Nedir?
- ✓ Kitapta neler öğreneceğiz?

IoT ile Yeni Nesil Robotik Uygulamalar

IoT Nedir? – Nesnelerin İnterneti ile Akıllı Sistemler

Günümüzde internet yalnızca bilgisayarlar ya da telefonlarla sınırlı değil. Artık evlerdeki lambalar, fabrikalardaki makineler, tarım alanlarında kullanılan sensörler hatta şehirlerdeki trafik lambaları bile internete bağlanabiliyor. Bu teknolojiye Nesnelerin İnterneti, yani İngilizce kısaltmasıyla IoT (Internet of Things) denir.

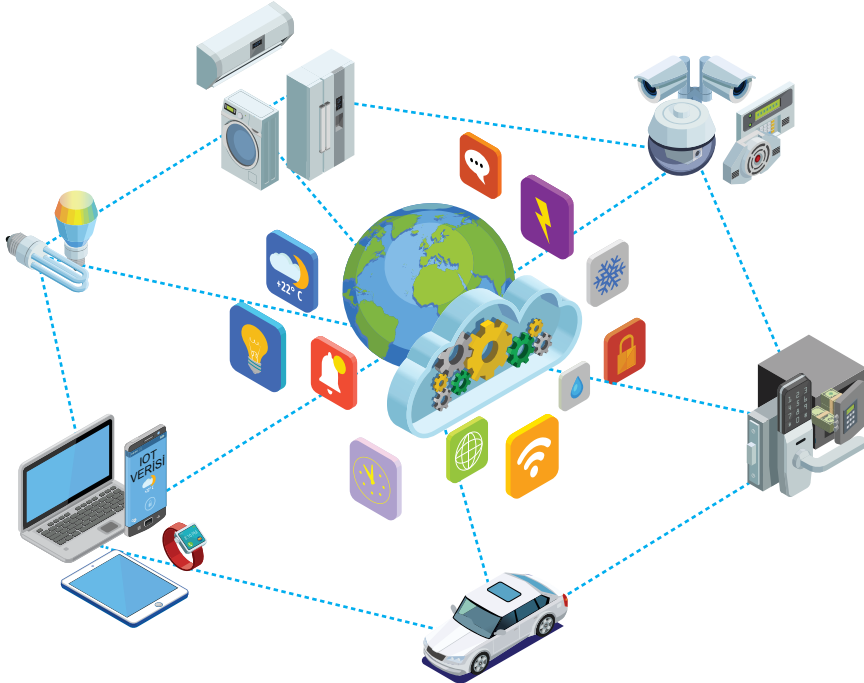
IoT, fiziksel nesnelerin internete bağlanarak veri topladığı, bu verileri paylaştığı ve hatta aldığı bilgilere göre bazı işlemleri otomatik gerçekleştirdiği sistemlerin genel adıdır. Örneğin bir sıcaklık sensörü ortam ısısını ölçebilir, bu bilgi uzaktaki bir bilgisayara gönderilebilir ve belli bir eşik değeri aşıldığında bir fan otomatik olarak çalıştırılabilir. Bu, küçük bir IoT senaryosudur.

IoT teknolojisi;

- ◆ Ev otomasyon sistemlerinden (ışık, ısıtma, güvenlik),
- ◆ Sanayi ve üretim hatlarına (Endüstri 4.0'ın önemli bir bileşeni olarak),
- ◆ Tarım uygulamalarından sağlık sistemlerine kadar

birçok alanda yaygın olarak kullanılmaktadır.

Bu kitap, lise düzeyindeki öğrencilerin IoT dünyasını tanımalarını ve temel uygulamalar geliştirerek bu teknolojiyi deneyimlemelerini amaçlamaktadır.



Bu Kitapta Neler Öğreneceğim?

IoT Eklenti Seti, Lise Laboratuvarı Ana Seti ile birlikte kullanıldığında, öğrencilere sensör verilerini toplama, yorumlama ve uzaktan erişimle sistem kontrolü yapma imkânı sunar. Öğrenciler, **ROBO Pro Coding** yazılımı ve **ROBOTICS TXT 4.0** kontrol ünitesi ile, grafiksel programlama ortamında konuya kolay bir giriş sağlar.

Bu kitap, IoT Eklenti Seti ile yapılabilecek **üç temel görev** üzerinden konuları ele almaktadır. Her görev, belirli bir probleme odaklanır ve öğrencilerin hem teorik bilgi edinmelerini hem de uygulama yapmalarını sağlar. Ayrıca her görev, ilgili sensörleri ve sistem bileşenlerini tanıtır, nasıl bağlantı kurulacağını ve hangi verilerin ölçüleceğini açıklar.

Kitaptaki Görevler

Görev 1: Hava Durumu İstasyonu

Bu görevde öğrenciler; sıcaklık, nem, hava basıncı, hava kalitesi ve aydınlık düzeyi gibi çevresel verileri ölçerek yorumlamayı öğrenir. Basınç ölçümlerine dayanarak **barometrik irtifa formülü** kullanılır ve basit bir hava durumu tahmini yapılır. Ayrıca **NTC sıcaklık sensörü** kullanılarak **Steinhart-Hart denklemi** ile sıcaklık hesaplaması yapılır. Tüm veriler, **dashboard** adı verilen kullanıcı arayüzü üzerinden görselleştirilir ve sisteme bir web kamerası da eklenebilir.

Görev 2: İç Mekan İklimi

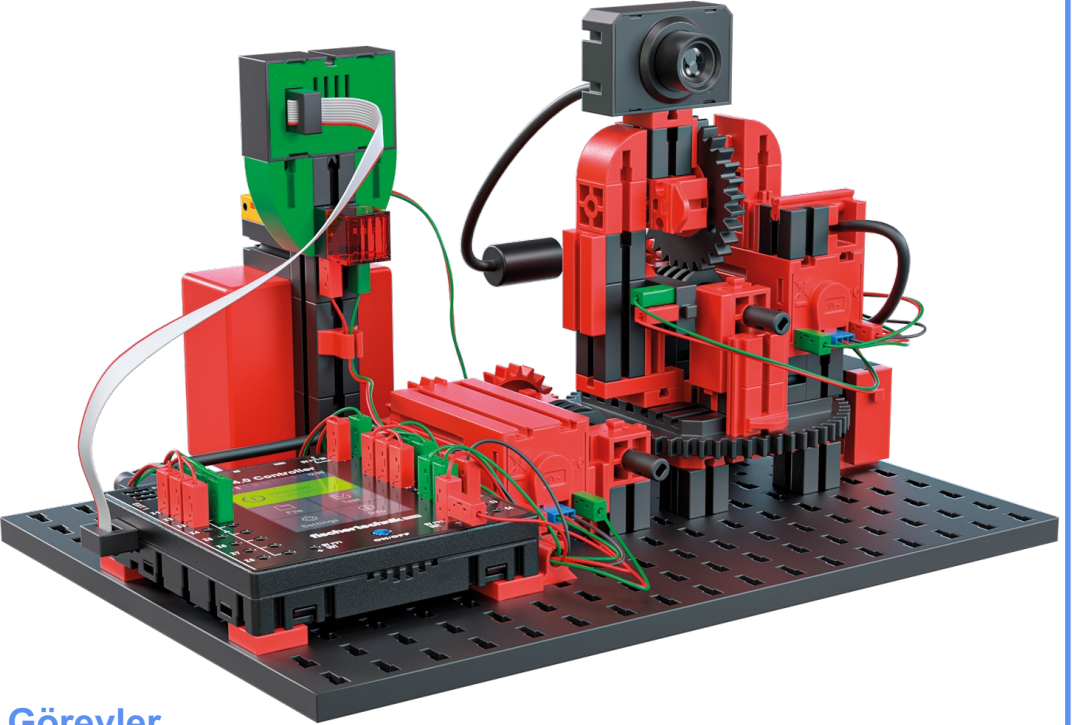
Bu görev, sınıf gibi kapalı alanlardaki hava kalitesini ölçmeye odaklanır. Öğrenciler; **bağıl nem, sıcaklık, hava kalitesi, aydınlık, ses seviyesi** gibi faktörleri sensörler aracılığıyla ölçer. Bu ölçümler doğrultusunda bir "**CO₂ trafik ışığı**" gibi havalandırma uyarı sistemi oluşturabilir. Ayrıca, **aydınlık ölçümü için foto direnç (LDR)** kullanımı ve **iklim konforu için ağırlıklı bir değerlendirme endeksi** hesaplama konuları da bu görevde ele alınır.

Görev 3: Alarm Sistemi

Bu görevde öğrenciler, hareket ve sesle tetiklenen bir **IoT tabanlı güvenlik sistemi** kurar. Sistemdeki kamera, iki eksenle hareket ettirilebilir (**pan-tilt kontrol**) ve uzaktan yönetilebilir. Web arayüzü üzerinden sisteme erişim sağlanır ve alarm sisteminin temel prensipleri öğrenilir. Öğrenciler aynı zamanda **encoder motorlarla hassas kontrol, fonksiyonlar kullanarak program yapısını düzenleme** ve **MQTT iletişim protokolü** ile tanışır.

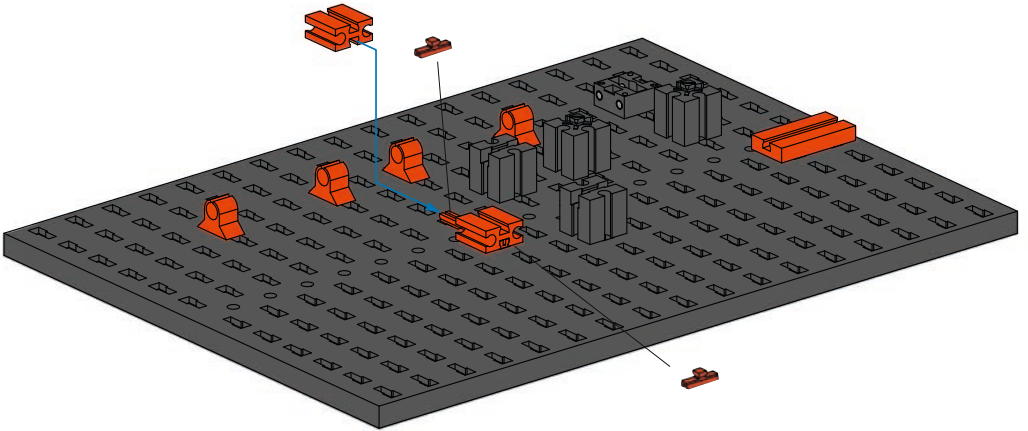
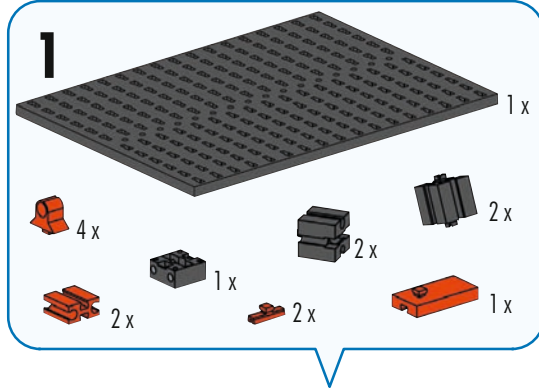
BÖLÜM 01

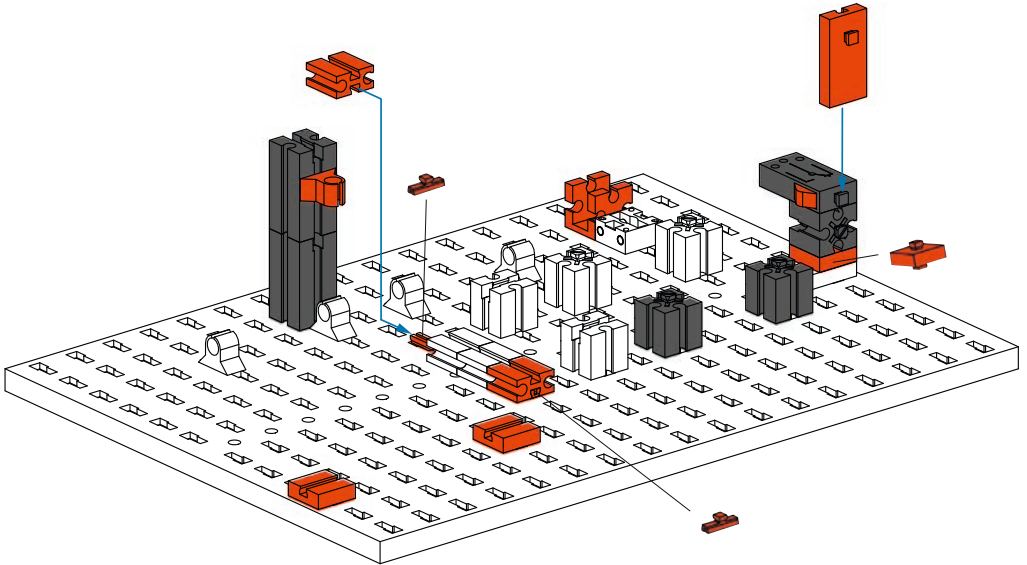
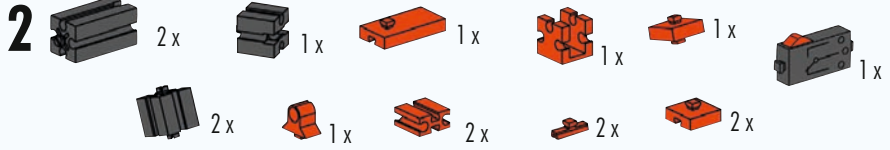
IoT Modeli

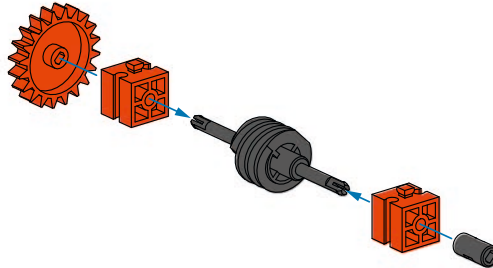
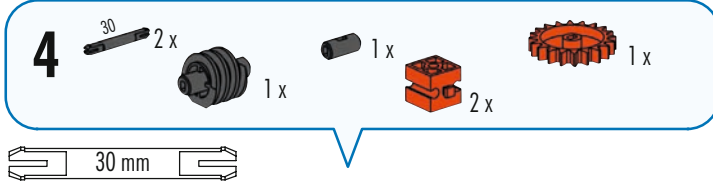
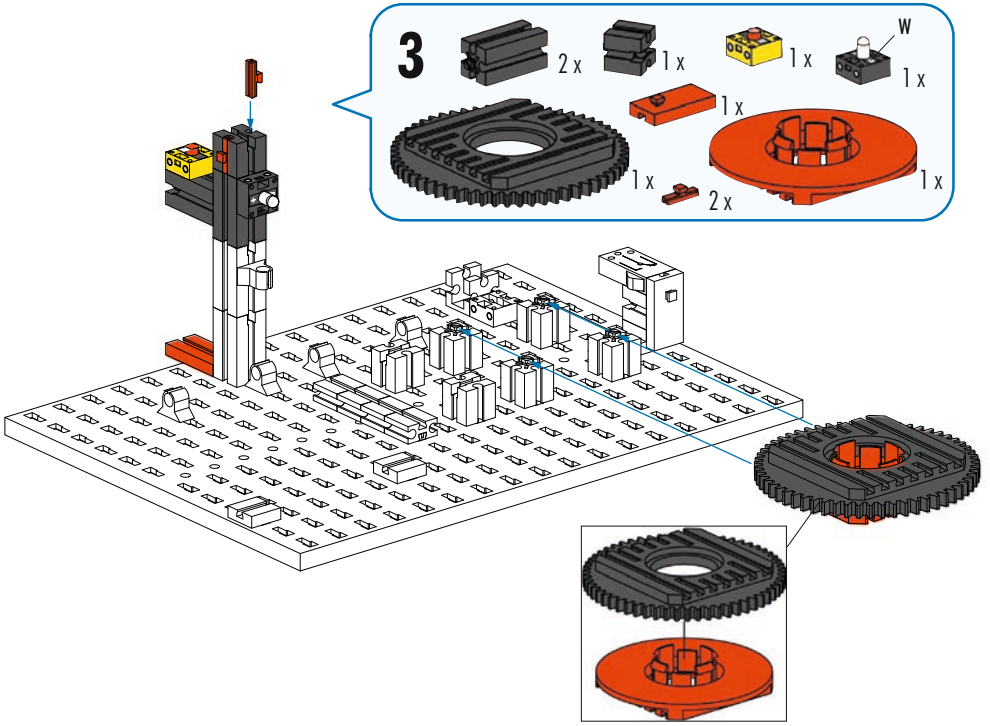


Görevler

- Hava Durumu İstasyonu
- İç Mekan İklimi
- Alarm Sistemi







5



4 x



2 x

15

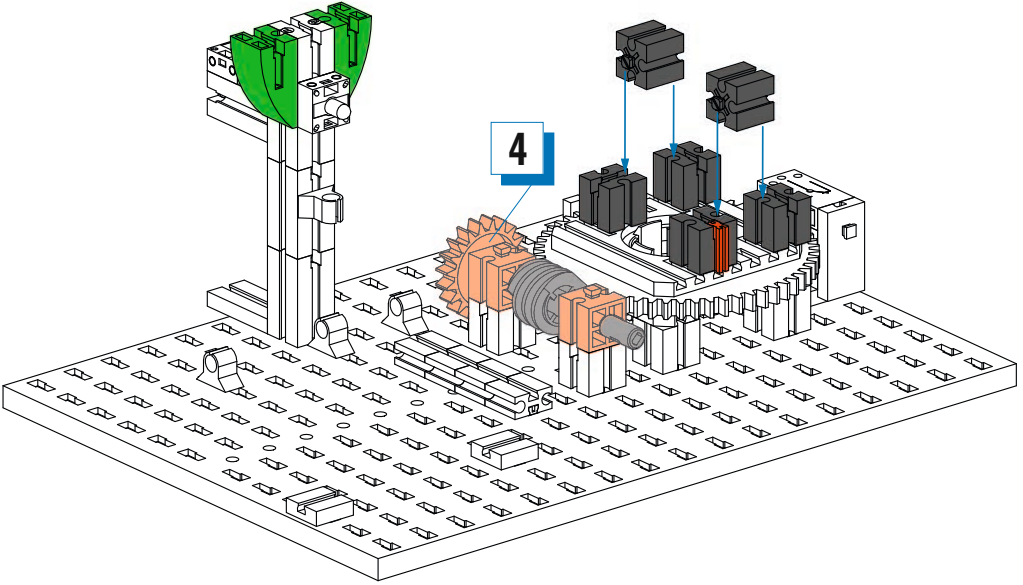


1 x

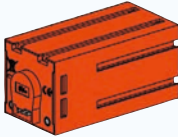


2 x

4



6



1 x



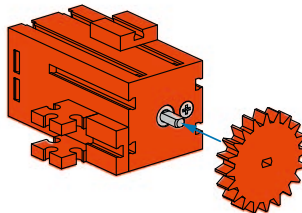
1 x



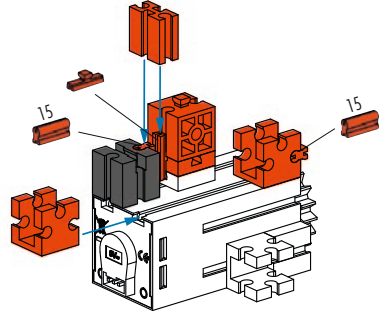
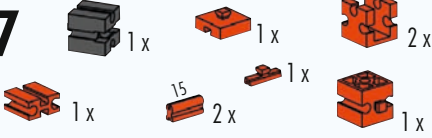
2 x



1 x



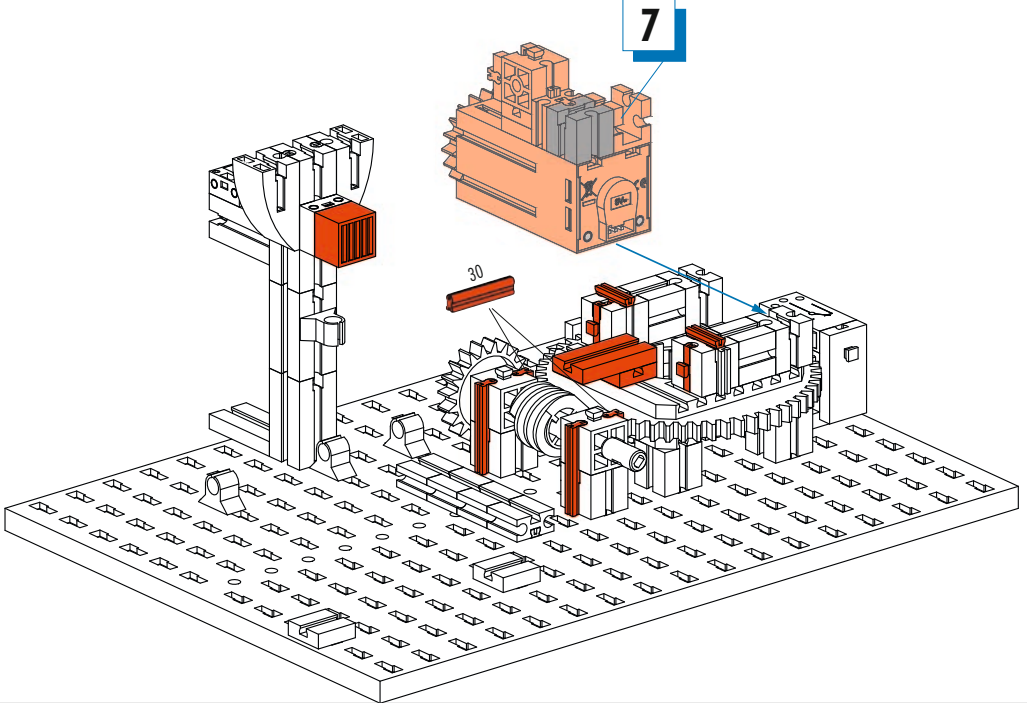
7



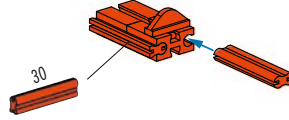
8



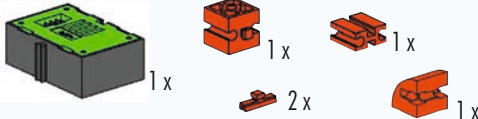
7



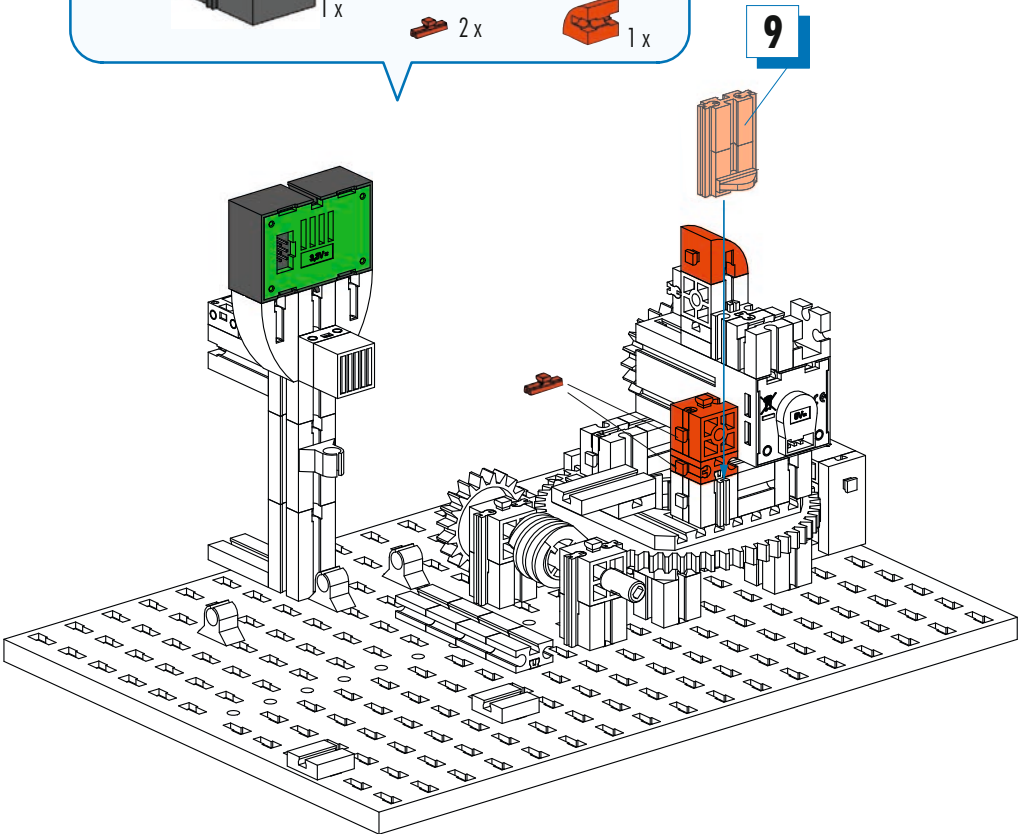
9

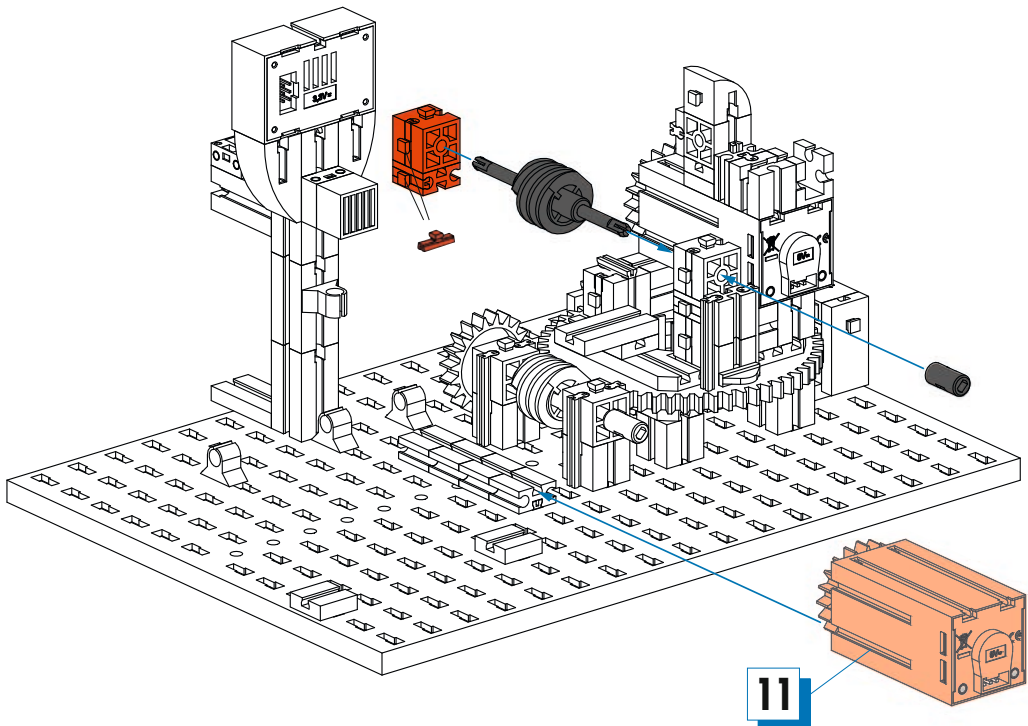
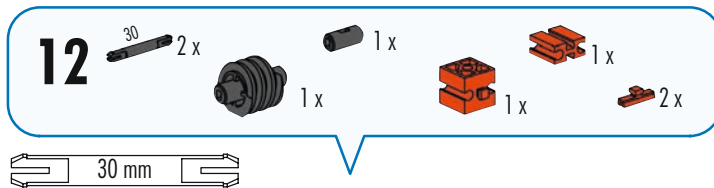
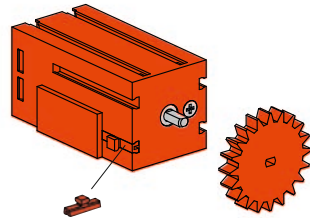
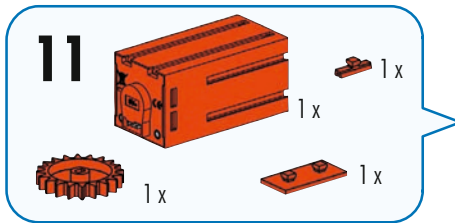


10



9





13



1 x

 $\frac{1}{x}$ 

1 x



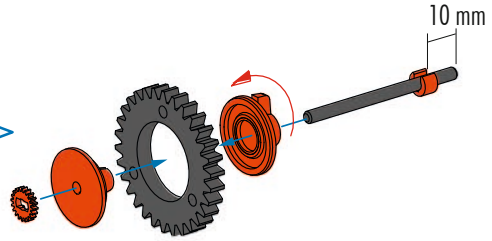
1 x



1 x



60 mm



14



1.



1.



1 x



1 x



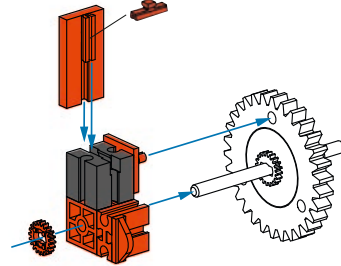
13



1 x



1 x



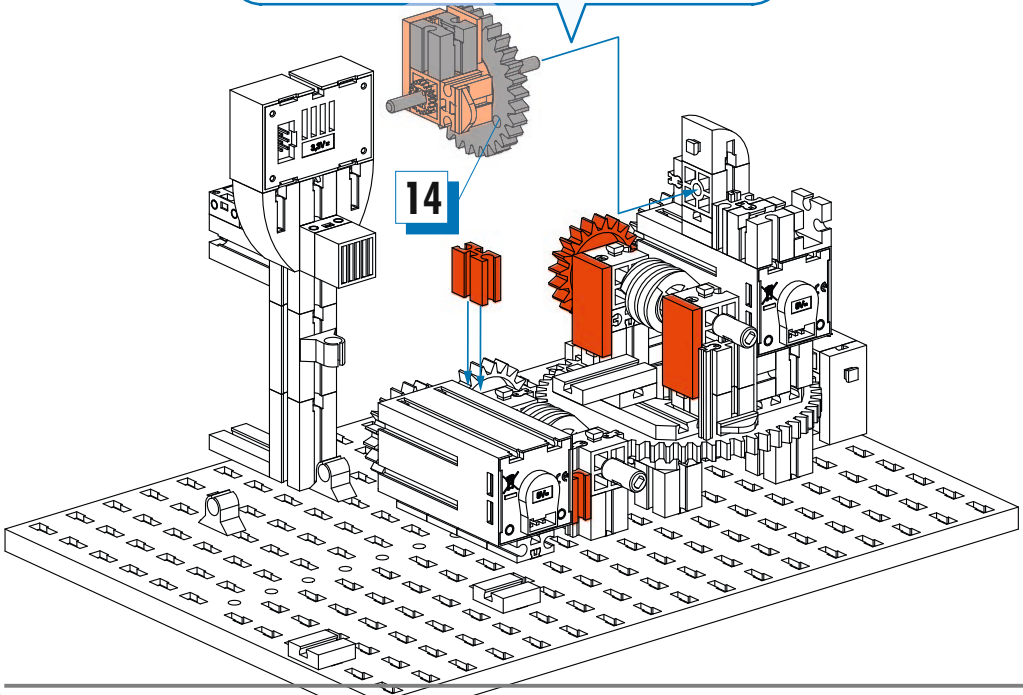
15



1)



2 x



16



1 x



1 x



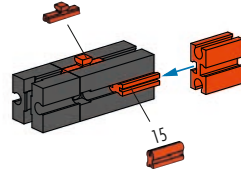
1 x



15 1 x



1 x



17



1 x



1 x



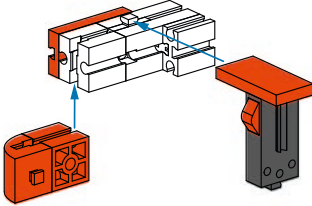
1 x



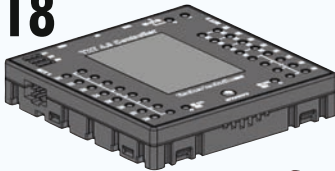
1 x



1 x



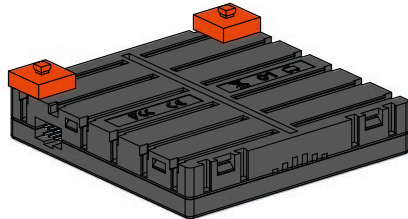
18



1 x



2 x



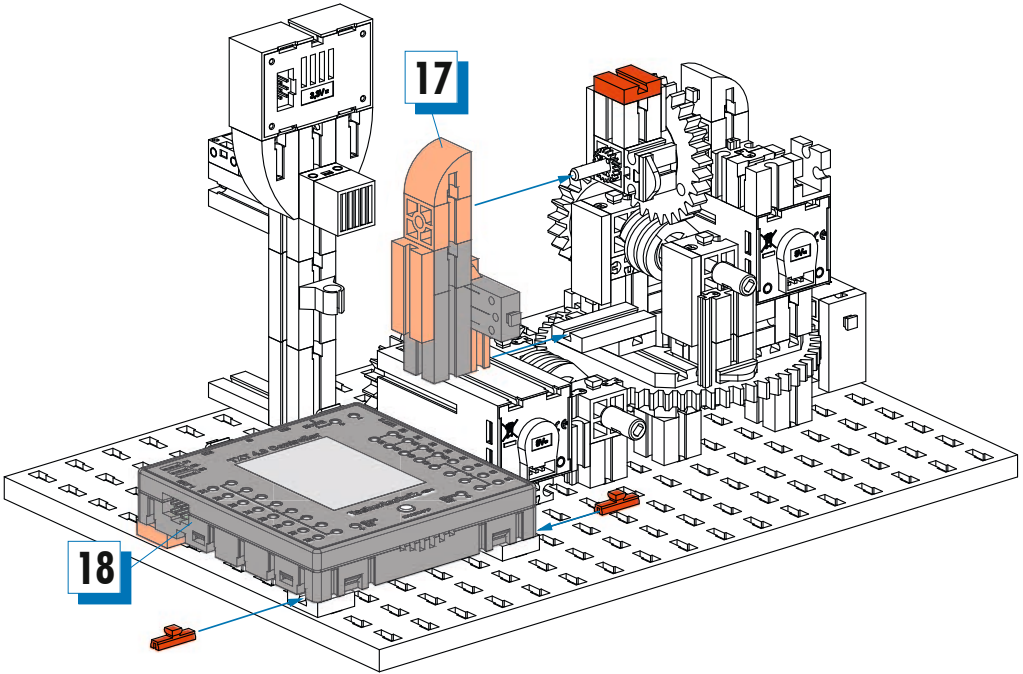
19



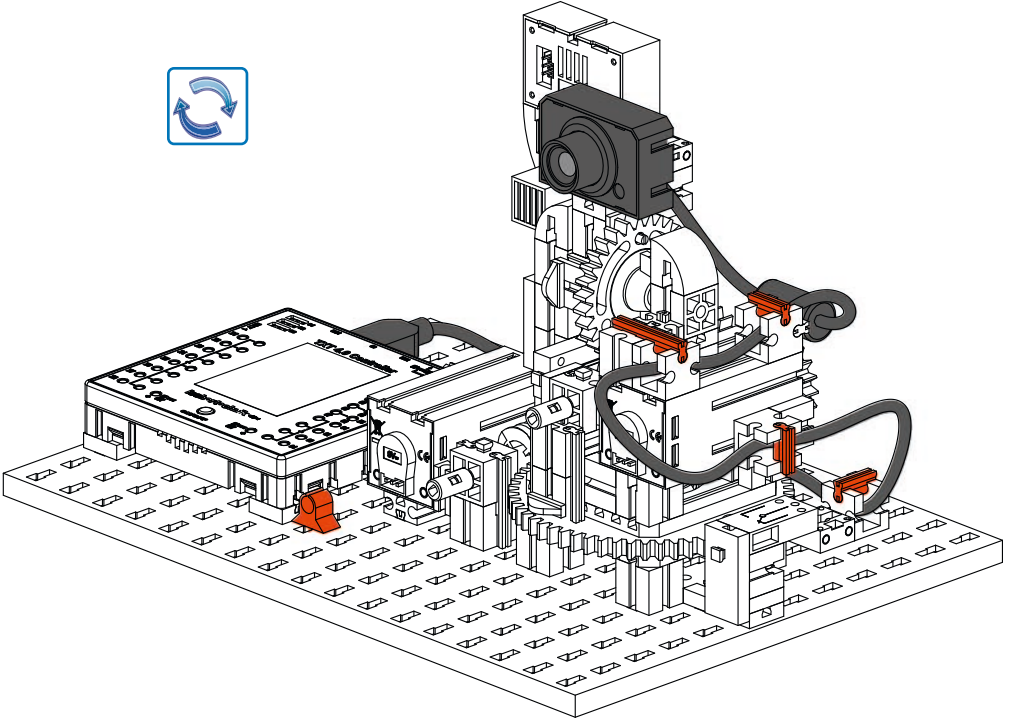
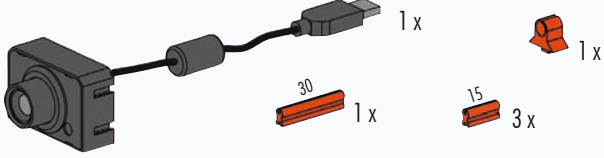
1 x



2 x



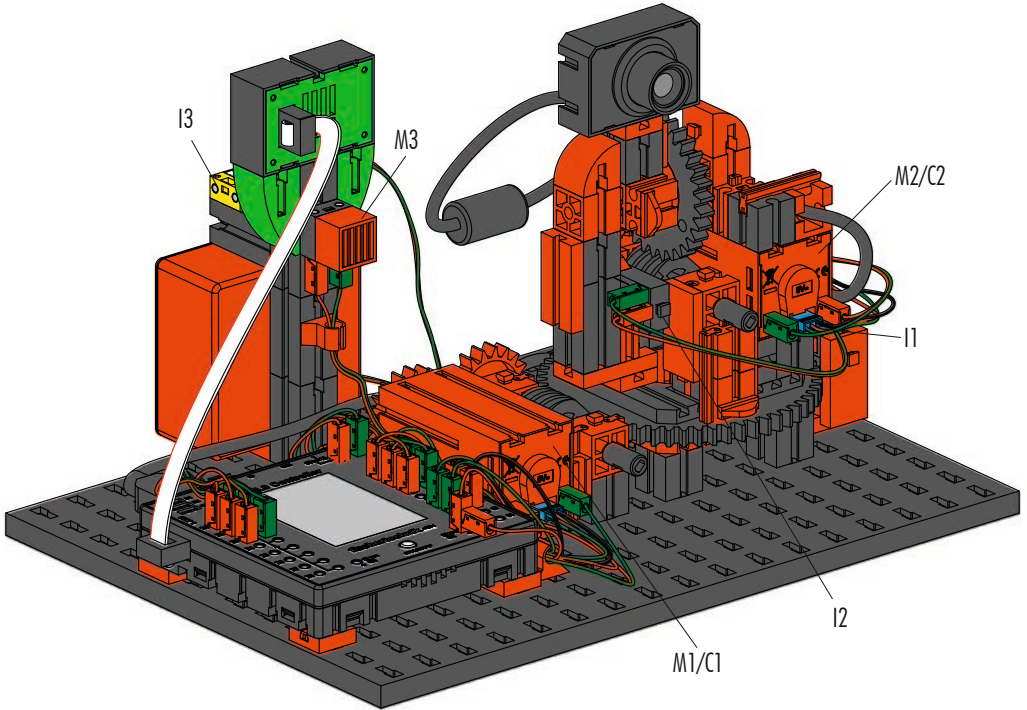
20



21

1 x 30 cm
1 x 60 cm

3 x 20 cm
4 x 30 cm
1 x 50 cm

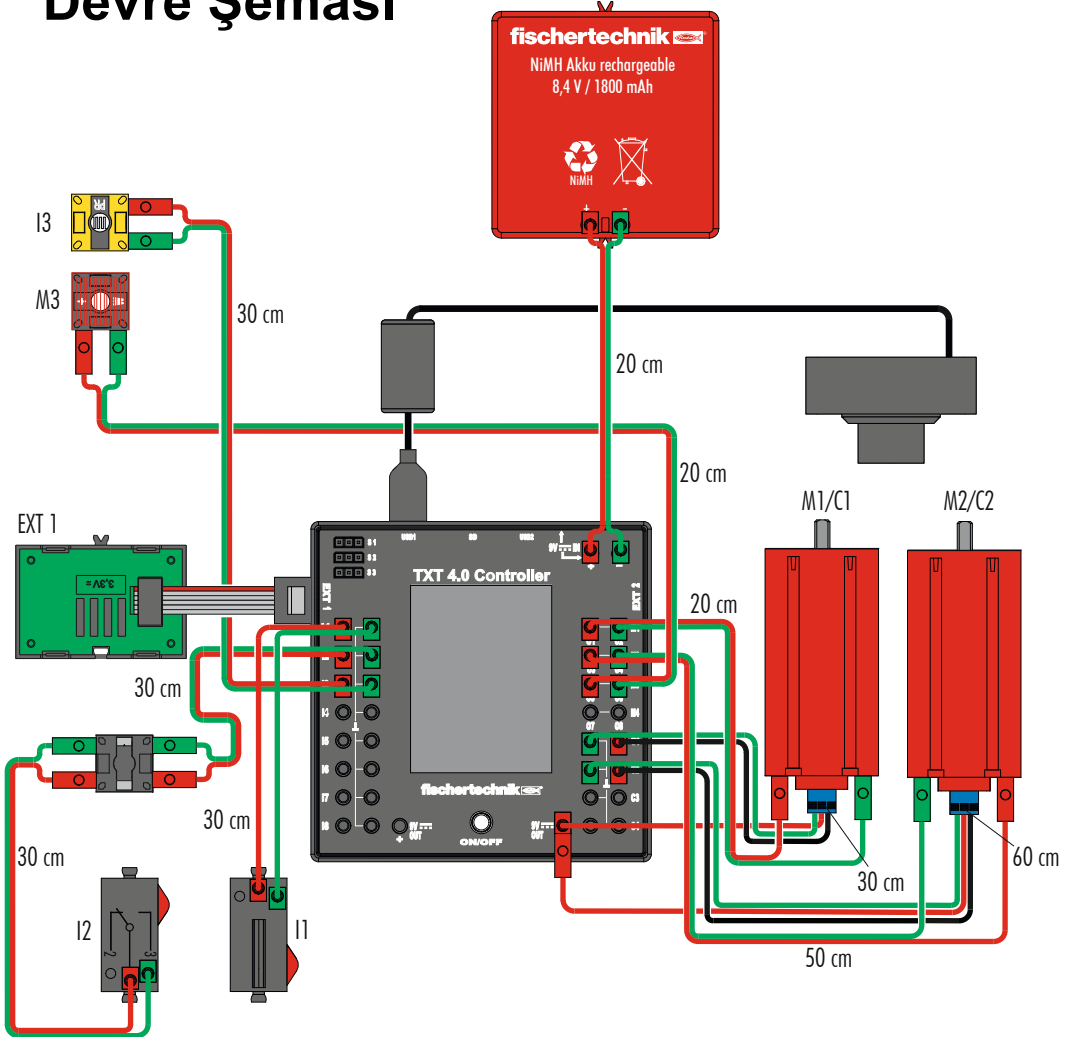


IoT Modeli

Devre Şeması



(+) ile gösterilen alana her zaman kırmızı soket takılır.





IoT Model İçin Öğrenme Hedefleri

Görev 1: Hava durumu istasyonu

Tüm sonraki görevler için bir model görevi göreceğ bir sensör istasyonu inşa ederek başlayacaksınız. Ardından, farklı hava durumu verilerini (nem, sıcaklık ve hava basıncı) toplamak, bu verilere dayanarak bir hava durumu tahmini hesaplamak ve verileri görselleştirme için sunucusundaki bir gösterge panosuna iletmek için çevresel sensörü (bir Bosch Sensortec BME680) kullanacaksınız.

Konular

- Hava durumunu görüntülemek için sensör verilerini analiz etme
- Hava basıncı ve barometrik formülü kullanarak hava tahmini
- Sensör verilerinin bir web sunucusunda görselleştirilmesi
- Hava durumu istasyonuna bir web kamerası ekleme

Öğrenme hedefleri

- Ölçülen değerlerin kaydedilmesi, anlamlı görselleştirme ve analiz
- Hava tahmini oluşturmak için barometrik formülü kullanma
- NTC direnci ve sıcaklık hesaplaması (Steinhart-Hart denklemi)

Gerekli malzemeler

- Program geliştirme için bilgisayar veya tablet.
- Programı TXT4.0'a iletmek için USB, BLE veya WiFi bağlantısı.
- Örnek programlar “IoT_MQTT.ft” and “IoT_Webcam.ft”

Daha fazla bilgi

- [1] Wikipedia: [Barometric formula](#).
- [2] fischertechnik: [NTC resistance](#), Data sheet, Art. no. 36437.
- [3] Stanford Research Systems (SRS): [Thermistor Calculator](#). V1.1

Görev 1: Hava durumu istasyonu

Bu görevde, sensör istasyonunu inşa ederek başlayacaksınız. Daha sonra, ölçülen verilerinizi MQTT protokolü (Mesaj Kuyruğu Telemetri Taşımacılığı) aracılığıyla bir web sunucusuna aktaracak bir hava durumu istasyonuna dönüştüreceğiz. Bir hava durumu tahmini tamamlayacak ve web kamerası işlevi görebilmesi için düzenli olarak bir kamera görüntüsü iletecektir.

Model Kurulumu

IoT istasyonunu yapı talimatlarına göre inşa edin. Sensörleri ve aktüatörleri kablo şemasına göre TXT 4.0'a bağlayın.

Deneyisel görevler sırasında, veriler MQTT protokolü kullanılarak bir web sunucusuna aktarılacak ve iletilen değerler orada görselleştirilecektir. Bunu yapmak için, fischertechnik cloud'ta bir hesap oluşturun (www.fischertechnik-cloud.com).

Programlama görevleri

1. Nem, sıcaklık ve hava basıncının ölçülmesi

Bosch'un BME680 tip tanımlamalı bir sensörü olan çevresel sensör, diğer özelliklerin yanı sıra bir sıcaklık, nem ve basınç sensörü içerir. Sensör değerleri I²C protokolü aracılığıyla TXT'ye iletilir. İlgili Blockly komutlarını kullanarak sensör değerlerini sorgulayabilirsiniz.

IoT istasyonunu, ölçülen nemi (% cinsinden), sıcaklığı (°C cinsinden) ve hava basıncını (hPa cinsinden) TXT'nin ekranına çıkaran ve bunları saniyede bir güncelleyen bir Blockly programı yazın.

2. Barometre

Ölçülen hava basıncını kullanarak hava istasyonuna bir tahmin fonksiyonu ekleyebilirsiniz: hava basıncı düşüyorsa, bu yağışın yakın olduğunu gösterir; hava basıncı artıyorsa, bu kuru ve güneşli havayı gösterir.

Deniz seviyesinde hava basıncı yaklaşık 950 ile 1050 hPa arasındadır; yaklaşık 1000 hPa ortalama değer civarında yaklaşık ± 50 hPa dalgalanır. Tüm ölçüm aralığını üç parçaya bölerseniz, ölçülen hava basıncını kullanarak bir tahmin yapabilirsiniz ("yağmurlu", "değişken", "kuru").

Ancak hava basıncı daha yüksek rakımlarda düşer. Bu nedenle, sensör tarafından ölçülen değerler farklı rakımlarda farklı şekilde yorumlanmalıdır. Bunu, "barometrik formül"e dayalı ölçülen hava basıncı $p(h)$ 'yi deniz seviyesindeki "teorik hava basıncı" p_0 'a dönüştürerek hesaba katabilirsiniz [1]:

$$p_0 = p(h) \cdot \left(\frac{T_0}{T(h)} \right)^{5.255} \text{ hPa}$$

Bu formülde,

- $p(h)$ yükseklikte ölçülen hava basıncı h ,
- $T(h)$ bu yükseklikte ölçülen sıcaklık (Kelvin cinsinden) ve T_0 deniz seviyesindeki (teorik) sıcaklık, her metre yükseklikle yaklaşık 0,0065 K düşer

$$T_0 = T(h) + 0.0065 \cdot h \frac{\text{K}}{\text{m}}$$

Hava istasyonunuzun deniz seviyesinden yüksekliğini h belirleyin ve bunu Blockly programınızda ölçülen hava basıncı $p(h)$ ve sıcaklık $T(h)$ temelinde "deniz seviyesindeki hava basıncını" hesaplamak için kullanın.

Daha sonra TXT ekranında gösterilecek hava durumu istasyonunuza bir hava durumu tahmini ekleyin.

Deneyisel görevler

1. NTC direncini kullanarak sıcaklığın belirlenmesi

Base Set için görev 1'de, Steinhart-Hart denklemini kullanarak sıcaklığı belirlemek için NTC direncini (termistör) kullandınız (eşlik eden materyallere bakın). Bunu yapmak için, önce termistörü kullanarak farklı sıcaklıklarda üç direnç değeri belirlemeniz gerekiyordu. Çevresel sensör şimdi bunları size yüksek bir doğruluk seviyesiyle sağlayacaktır.

NTC direncini I8'e bağlayın ve NTC direncinin bir ölçümünü Blockly programınıza ekleyin. Üç ölçümün sonuçlarını aşağıdaki tabloya girin:

Deniz seviyesinde hava basıncı yaklaşık 950 ile 1050 hPa arasındadır; yaklaşık 1000 hPa ortalama değer civarında yaklaşık ± 50 hPa dalgalanır. Tüm ölçüm aralığını üç parçaya bölerseniz, ölçülen hava basıncını kullanarak bir tahmin yapabilirsiniz ("yağmurlu", "değişken", "kuru").

Direnç değeri	Sıcaklık

Not: Çevresel sensör, gövdeden dolayı sıcaklık değişikliklerine uzun bir gecikmeyle tepki verecektir. Bu nedenle, yalnızca sensörün sıcaklık değeri sabitlendiğinde ölçümü tamamlayın.

Artık Steinhart-Hart denkleminin katsayılarını, ekteki [4] numaralı dokümanda belirtilen eşlik eden materyalleri ve web sitesini kullanarak belirleyebilir ve ayrıca NTC direncinden hesaplanan sıcaklığı TXT'nin ekranında çıktı olarak verebilirsiniz.

2. IoT sunucusunda veri görselleştirme

Şimdi, hava istasyonunuzdan gelen ölçüm verileri bir web sunucusuna iletilmeli ve orada (hazırlanmış) bir gösterge panelinde görüntülenmelidir. Bunu yapmak için, TXT'nizi daha önce fischertechnik bulutunda kurduğunuz hesaba bağlayın.

2a. Şimdi, panoyu sıcaklık, nem ve hava basıncının görüntüleneceği şekilde yapılandırın. Seçimlerini kaldırmanız gerekmeyen tüm ekranlara tıklayın.

2b. Aşağıdaki örnek program, sensör verilerinin IoT sunucusuna nasıl iletildiğini göstermektedir: Öncelikle bulut hesabınızı kullanarak bağlantıyı açın ("MQTT connect"); ardından, sensörden gelen mevcut ölçülen değerler düzenli aralıklarla sabit bir formatta (zaman damgası, sıcaklık, nem, basınç) bulut sunucusuna iletilir ("MQTT publish").

```

program start
  fischertechnik cloud connect
  wait until is connected to fischertechnik cloud
  repeat forever
    do fischertechnik cloud publish text to /j1/text/<txid>/i/bme680 - format text - {"ts":"","t":"","h":"","p":""}
    + - with - timestamp
    + - get environmental sensor TXT_M_1 - temperature
    + - get environmental sensor TXT_M_1 - humidity
    + - get environmental sensor TXT_M_1 - pressure
  wait 5 - 1

+ define timestamp
python code
  ts = datetime.now().strftime("%Y-%m-%dT%H:%M:%S...")
  return ts

python imports
  import json
  from datetime import datetime

```

IoT_MQTT.ft

Barometre için IoT sunucusundaki ekranı Blockly programınıza ekleyin.

Artık ölçüm serileri oluşturabilir, ölçülen değerleri görüntüleyebilir ve bunları panoya csv dosyası olarak indirebilirsiniz. Csv dosyasını bir elektronik tabloda (örneğin Excel'de) analiz edebilir ve düzenleyebilirsiniz.

3. Web kamera

Web kamerası olmayan bir hava durumu istasyonu ne işe yarar? Programınızı, kamera görüntüsünü IoT sunucusunun panosuna iletecek şekilde genişletin.

3a. Şimdi, kamera görüntüsünün bir gösterimini gösterge paneline ekleyin.

3b. Aşağıdaki örnek program, kamera görüntüsünün iletiminin nasıl çalıştığını gösterir: Kameranın "çektiği" her görüntü Base64 biçiminde kodlanır. Ardından, geçerli görüntü saniyede bir kez IoT sunucusuna iletilir ("MQTT yayını").

```

program start
  set snapshot - to - " "
  fischertechnik cloud connect
  wait until is connected to fischertechnik cloud
  repeat forever
    do fischertechnik cloud publish text to /j1/text/<txid>/i/cam - format text - {"ts":"","data":""}
    + - with - timestamp
    + - snapshot -
  wait 5 - 1

on image TXT_M_USB1_1 - change: event
  set snapshot - to - convert image - event - to base64

+ define timestamp
python code
  ts = datetime.now().strftime("%Y-%m-%dT%H:%M:%S...")
  return ts

python imports
  import json
  from datetime import datetime

```

IoT_Webcam.ft

Programınızı, kamera görüntüsünü IoT sunucusuna iletecek şekilde genişletin.

Panoyu kullanarak anlık görüntüleri bir galeride kaydedebilir ve görüntüleri tek tek indirebilirsiniz.



IoT Model İçin Verilen Çözümler

Çözüm sayfası - Hava durumu istasyonu (Görev -1)

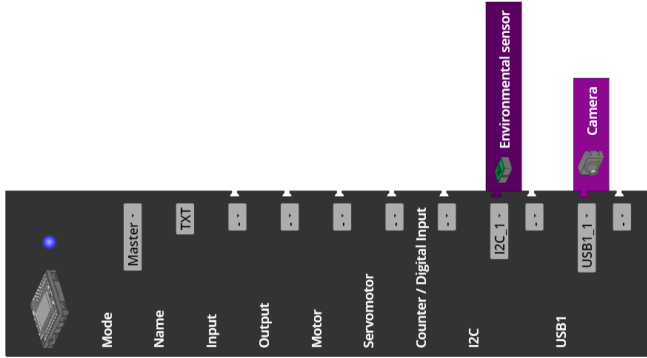
Görev 1 ve 2, sensör istasyonunu yerel bir hava istasyonuna dönüştürür. Deneysel görevlerde, hava istasyonu, ölçülen verileri ve iletilen görüntüleri bir gösterge panelinde görüntüleyen MQTT aracılığıyla bir web sunucusuna bağlanır.

Modelin Kurulumu

Yapım talimatlarına bakınız.

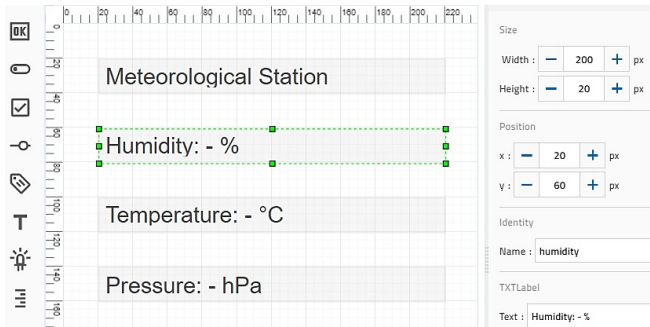
Programlama görevleri

Sensör ve aktüatörlerin yapılandırılması:



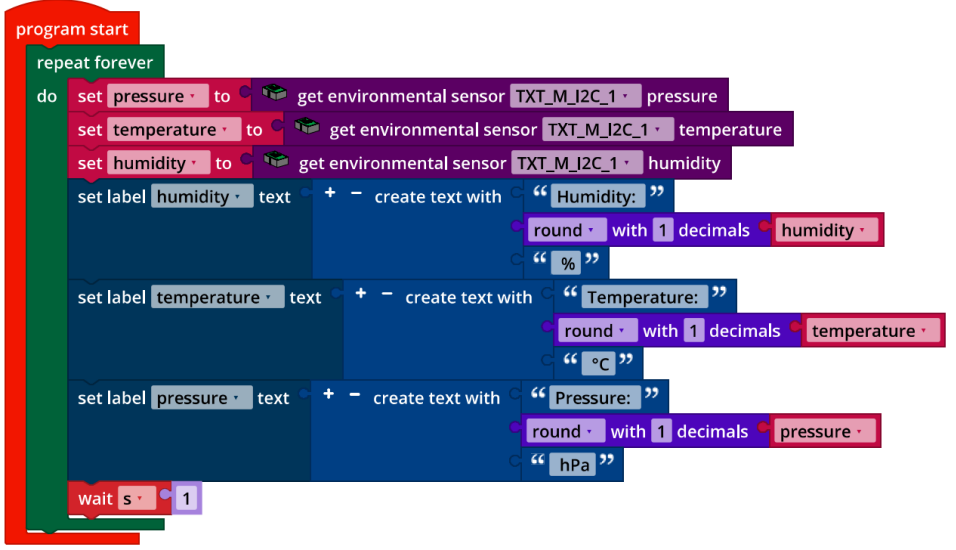
1. Nem, sıcaklık ve hava basıncının ölçülmesi

1a. Görüntü yapılandırması (örnek):



Görüntü çıktısını yapılandırma

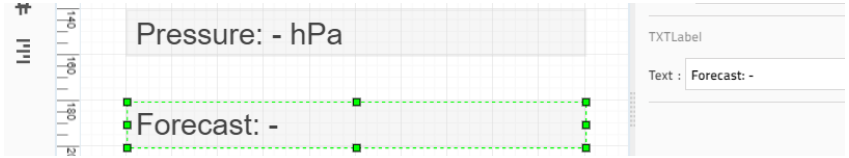
1b. Program (örnek):



IoT_Meteorological_Station.ft

2. Barometre

2a. Ekran yapılandırmasının genişletilmesi (örnek):

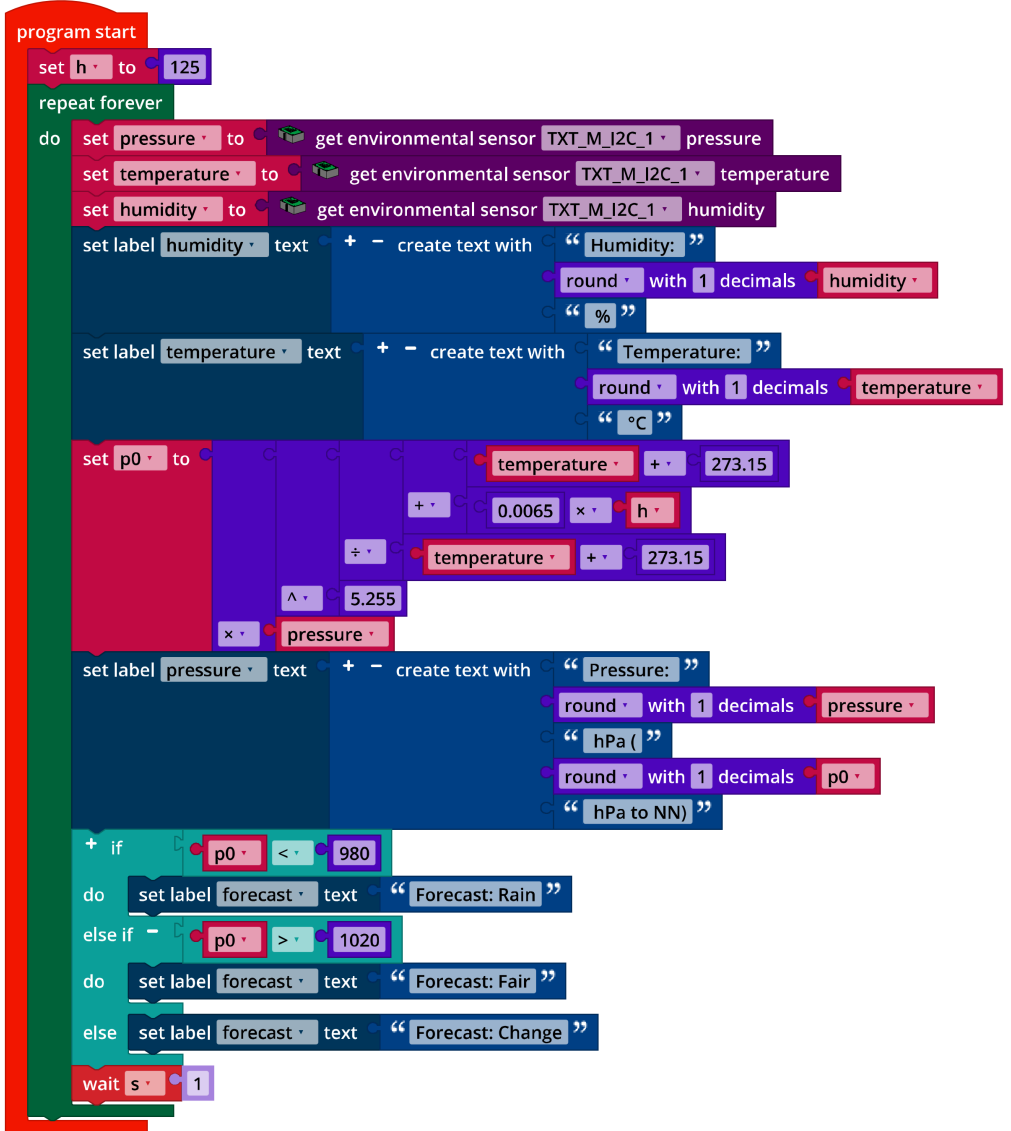


Görüntü çıktısını genişletme

2b: Program (örnek):

Hesaplamadan önce sıcaklık değerlerinin Kelvin'e dönüştürülmesi gerekir:

$$0^{\circ}\text{C} = 273.15 \text{ K.}$$



IoT_Barometer.ft

Deneysel görevler

1. NTC direncini kullanarak sıcaklığın belirlenmesi

1a. Termistörün direnci ve çevre sensörü için ölçülen sıcaklık değerleri konsoldaki Log çıkışı aracılığıyla kolayca belirlenebilir (aşağıdaki örnek programa bakın).

Steinhart-Hart denkleminin [4, 5] katsayıları a, b ve c'yi belirlemek için üç ölçülen değer kullanılabilir (T: Kelvin cinsinden Sıcaklık, R: Ohm cinsinden NTC direnci):

$$\frac{1}{T} = a + b \cdot \ln(R) + c \cdot \ln(R)^3$$

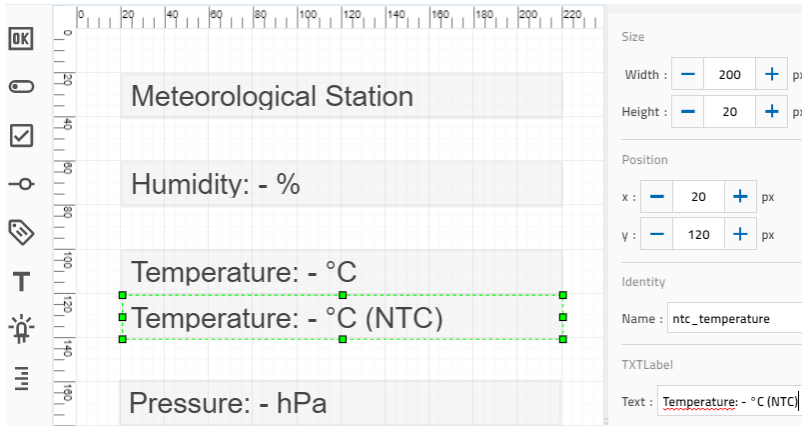
Direnç değeri	Sıcaklık
1500 Ω	25.0 °C
2362 Ω	13.7 °C
2530 Ω	11.8 °C

Örnek ölçüm

Ölçülen değerler NTC direncine bağlı olup sapma gösterebilir.

Katsayılar: a = 0.004535418128, b = -0.0003767491105, c = 0.000004023802790

1b. Ekranı genişletme:

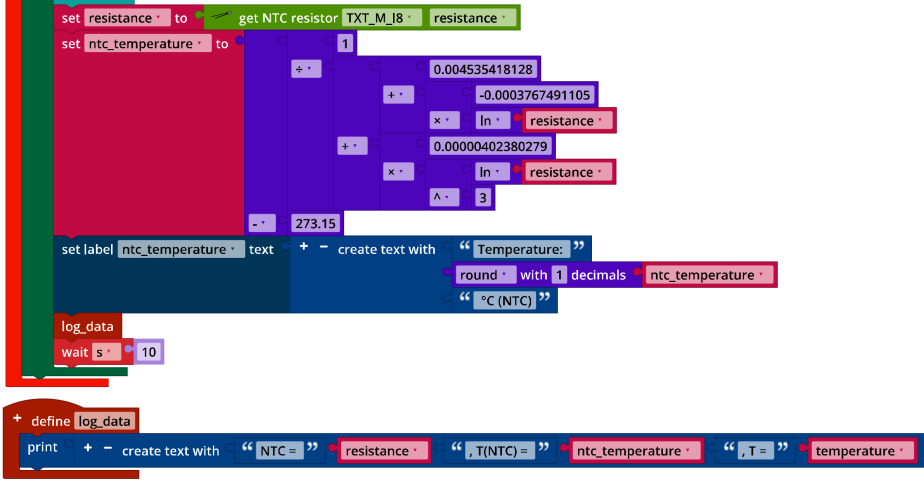


NTC sıcaklık ölçümünün ekran çıktısının yapılandırılması

1c. Programın genişletilmesi (örnek):

Konsoldaki NTC direnci ve sıcaklık sensörü için ölçülen değerlerin çıkışı ve NTC direncine göre sıcaklık değerinin hesaplanması.

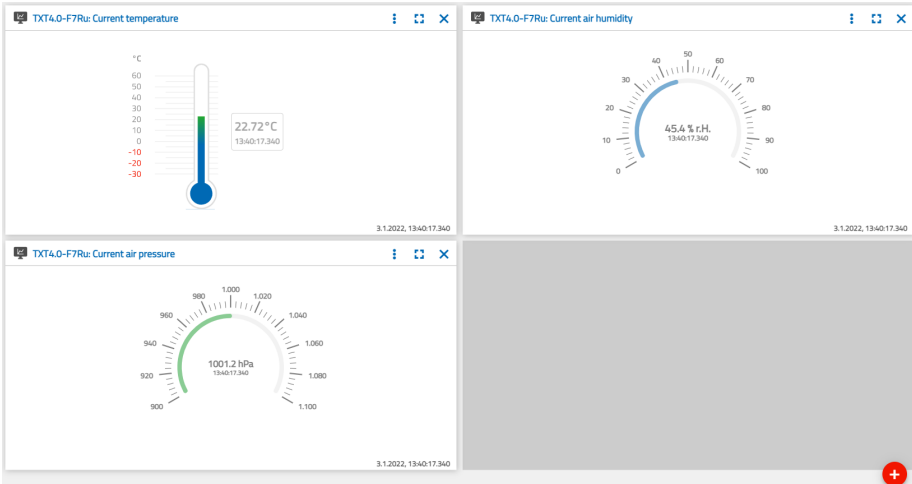
Burada da termistörden hesaplanan sıcaklık değerinin Kelvin'e çevrilmesi gerekir:
 $0^{\circ}\text{C} = 273.15 \text{ K}$.



IoT_Barometer_with_NTC_resistor.ft

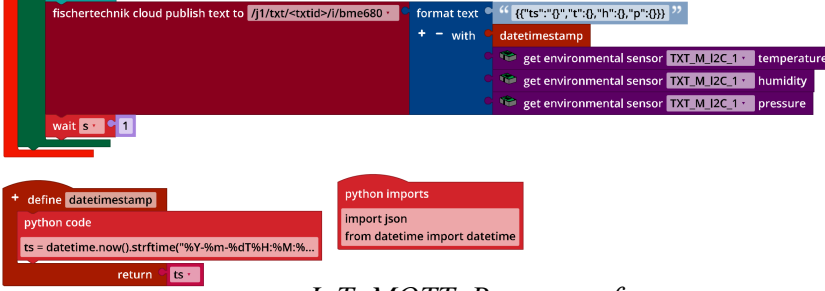
2. IoT sunucusunda veri görselleştirme

2a. Fischertechnik bulutundaki IoT sunucusundaki panonun yapılandırılması:



Gösterge panelindeki görüntülemeyi yapılandırma

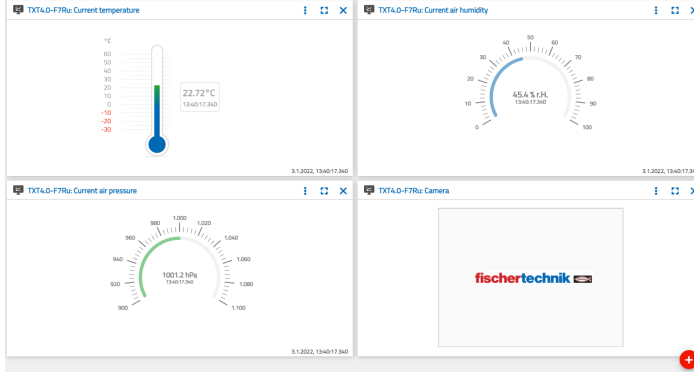
2b. Programın genişletilmesi (örnek):



IoT_MQTT_Barometer.ft

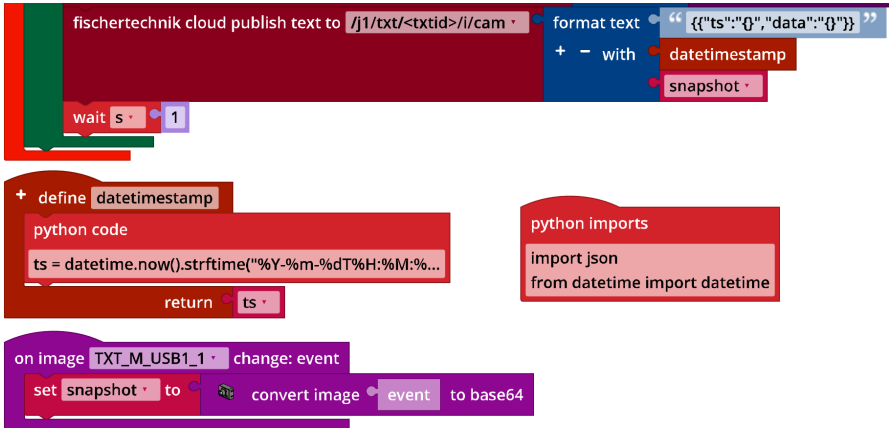
3. Web kamerası

3a. Fischertechnik bulutundaki gösterge panelinin genişletilmesi:



Web kamerası olan gösterge paneli

3b. Programın genişletilmesi (örnek):



IoT_MQTT_Barometer_with_Webcam.ft



IoT Model İçin Öğrenme Hedefleri

Görev 2: İç mekan iklimi

Sensör istasyonu iç mekan iklimini analiz etmek için kullanılacak ve belirtilen sınır değerleri aşıldığında veya karşılanmadığında bir uyarı verecektir. Ölçüm serileri IoT sunucusu kullanılarak toplanabilir ve ardından analiz edilebilir.

Konular

- İç mekan iklimini etkileyen faktörlerin ölçülmesi (bağıl nem, sıcaklık, hava kalitesi, parlaklık, hacim)
- Sınıflar için bir "havalandırma göstergesi" tanımlamak ve ölçmek ("CO2 trafik ışıkları")
- Aydınlatma seviyesinin belirlenmesi

Öğrenme hedefleri (Bağıl nem terimini anlamak)

- Gürültü seviyesinin (desibel) nasıl ölçüleceğini anlamak
- Aydınlatma seviyesi (lüks) terimini anlamak ve bu değeri bir fotodirençten alınan değerlere göre belirlemek
- Bağıl nem, sıcaklık, hacim ve aydınlatma seviyesi için tipik sınır değerlerini öğrenme
- Belirtilen bir ağırlığa dayalı olarak birden fazla ölçülen değerden bir endeks değeri türetme

Daha fazla bilgi

- [1] Bosch Sensortec: BME680 – Application Note. Rev. 1.6, 20.09.2020.
- [2] Bosch Sensortec: BME680 – Data sheet. Rev. 1.7, 20.12.2021.
- [3] fischertechnik: Photoresistor LDR03 (32698). Data sheet, 17.10.2018.
- [4] German Federal Ministry of Labour and Social Affairs: Beleuchtung. Technische Regeln für Arbeitsstätten (Lighting: technical work station regulations), ASR A3.4, April 2011.

Görev 2: İç mekan iklimi

Bu görevde, sensör istasyonumuz iç mekan iklimini izleyecek ve çok sıcak, çok nemli, çok karanlık veya çok gürültülü olup olmadığını veya hava kalitesinin düşüp düşmediğini bize bildirecek (örneğin aşırılık nedeniyle CO2).

Model Kurulumu

IoT modeli üzerinde herhangi bir değişiklik yapmadan kullanabilirsiniz.

Programlama görevleri

1. İç mekan iklim değerlerinin ölçülmesi

Görev 1'de toplanan "nem" ve "sıcaklık" ölçüm değerlerini yalnızca hava durumunu değil, aynı zamanda iç mekan iklimini belirlemek için de kullanabilirsiniz.

Nem ve sıcaklık iyi bir iç mekan iklimi için önemli faktörlerdir. (Mutlak) nem, bir metreküp (m³) havada su buharı olarak bulunan su miktarını (g cinsinden) gösterir.

Sensör bağlı nemi veya (mutlak) nemin ve mümkün olan maksimum nemin % cinsinden oranlarını ölçer. Maksimum nem sıcaklıkla değişir: 0°C'de hava en fazla 5g su buharı emebilir, ancak 30°C'de 30g emebilir - bu nedenle soğuk hava sıcak havadan daha "kurudur".

Yaşam ve çalışma alanları için optimum iç ortam iklimi, 20°C sıcaklıkta %45-55 bağlı nem olarak kabul edilir.

1a. Görev 1'deki hava istasyonunun görüntü yapılandırmasını, bağlı nem ve sıcaklığı gösterecek şekilde ayarlayın.

1b. Hava istasyonunun Blockly programını değiştirerek onu bir iç mekan iklim istasyonu haline getirin.

Not: Program başlatıldıktan sonra ölçülen değerlerin sabitlenmesi için sensörün yaklaşık 10 saniyeye ihtiyacı olacaktır [1].

2. Hava kalitesinin ölçülmesi

Çevresel sensör, formaldehit, alkol, çözücüler ve buharlaşan vernikler, lekeler, temizlik maddeleri veya yapıştırıcılar gibi ortam havasındaki çeşitli gazları tespit edebilir. Bunları kullanarak 0-500 aralığında bir Hava Kalitesi Endeksi, IAQ belirleyebilir.

50'nin altındaki bir IAQ değeri iyi hava kalitesini gösterirken, 200'ün üzerindeki bir değer havada önemli bir kirliliğin olduğunu gösterir (tabloya bakınız).

IAQ Index	Air Quality	Impact (long-term exposure)	Suggested action
0 – 50	Excellent	Pure air; best for well-being	No measures needed
51 – 100	Good	No irritation or impact on well-being	No measures needed
101 – 150	Lightly polluted	Reduction of well-being possible	Ventilation suggested
151 – 200	Moderately polluted	More significant irritation possible	Increase ventilation with clean air
201 – 250 ⁹	Heavily polluted	Exposition might lead to effects like headache depending on type of VOCs	optimize ventilation
251 – 350	Severely polluted	More severe health issue possible if harmful VOC present	Contamination should be identified if level is reached even w/o presence of people; maximize ventilation & reduce attendance
> 351	Extremely polluted	Headaches, additional neurotoxic effects possible	Contamination needs to be identified; avoid presence in room and maximize ventilation

Hava Kalitesi Endeksi [2]

2a. İç mekan iklim istasyonunun ekran yapılandırmasını, hava kalitesini gösterecek şekilde ayarlayın.

2b. Buna göre Blockly programını iç mekan iklim istasyonuna yönelik olarak genişletin.

Not: Program başlatıldığında hava kalitesi sensörü otomatik olarak kalibre edilir. Kalibrasyon işlemi beş dakika sürecektir ve ilerleme konsolda gösterilecektir. Bu süre zarfında, IAQ değeri olarak -1 gösterilecektir.

Deneyisel görevler

1. Parlaklık

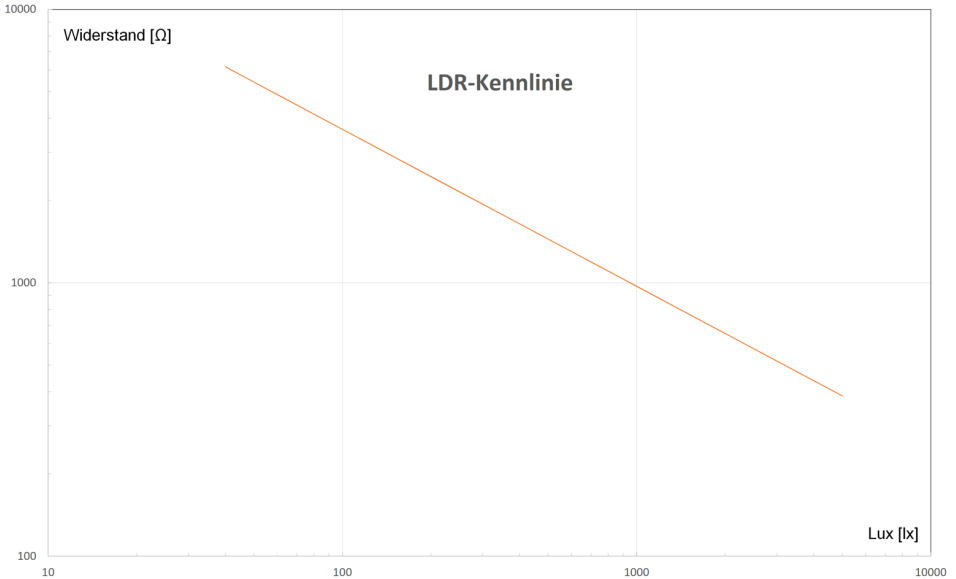
Robotik Temel Setinin 2. görevinde ışık bariyerinde sensör olarak kullanılan fototransistörün aksine, bir fotodirenç (LDR – Light Dependent Resistor) dijital bir sensör değil, analog bir sensördür. Bunu ortam parlaklığını belirlemek için kullanabiliriz.

Lüks (lx), aydınlatma seviyesi birimidir - belirli bir yüzeye çarpan ışık akısı (lümen cinsinden ölçülür). Aşağıdaki formül geçerlidir: 1 Lüks = 1 Lümen/m². Bir lüks, bir mum alevinden bir metre uzaklıkta ölçülebilen aydınlatma seviyesine karşılık gelir.

Karşılaştırma yapmak gerekirse: Bulutsuz bir yaz gününde, doğrudan güneş ışığı altındaki aydınlatma seviyesi 100.000 lükse ulaşabilirken, bulutlu bir kış gününde bu seviye yalnızca 2.000 lüks civarında olacaktır.

fischertechnik fotodirenç 60 kOhm'dan fazla direnç değerleri sunar; ne kadar parlaksa direnç o kadar düşüktür. 1 kOhm'luk bir direnç değeri yaklaşık 1000 lux'e karşılık gelir, buna "aydınlatma altındaki direnç" denir).

Direnç için karakteristik eğri, direnç değeri ile aydınlatma seviyesi arasındaki ilişkiyi gösterir. Logaritmiktir (ölçeklere dikkat edin):



1a. Şimdi, birden fazla ölçümü tamamlamak için bir ışık ölçer (örneğin bir uygulama) kullanın ve direnç değerini ve ölçülen lüks değerini bir tabloya girin. Ölçülen değerler için basit bir formül elde etmek için bir elektronik tablo programı kullanabilirsiniz.

of $y = a \cdot xb$, Direnç değerine göre yaklaşık aydınlatma seviyesini lüks cinsinden hesaplamak için kullanabilirsiniz.

1b. Programlama görevi 2'den Blockly programına aydınlatma seviyesinin lüks cinsinden gösterimini ekleyin.

1c. Sınıftaki minimum aydınlatma seviyesi için uygun bir eşik değeri tanımlayın. Örneğin, işyeri yönetmelikleri ne gerektirir [4]? İkinci LED'i O7 ve O8'e bağlayın (doğru polariteyi sağlayın). Oda çok karanlık olursa Blockly programınız tarafından açılmalıdır.

2. CO2 trafik ışıkları

Çevresel sensör havadaki CO2 seviyesini doğrudan belirleyemez, ancak içinde bulunan diğer gazları kullanarak insanların soluduğu havayı tespit edebilir. Bu nedenle, sensör tarafından ölçülen hava kalitesi doğrudan nefes verme sonucu havadaki CO2 içeriğine bağlıdır.

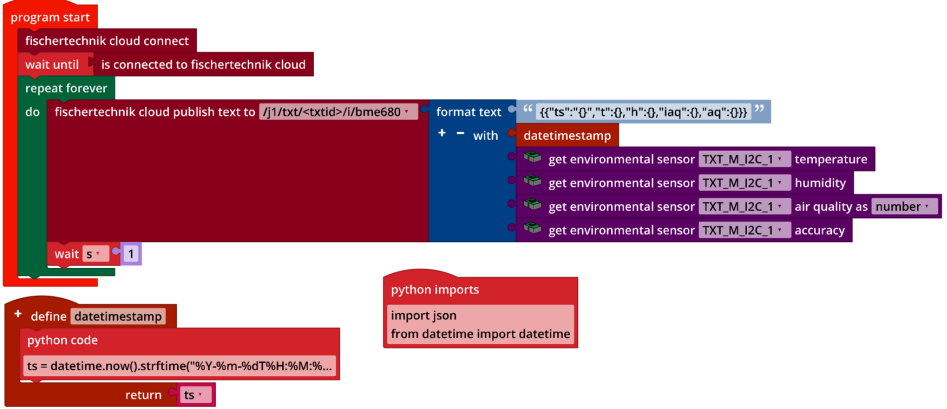
Havadaki CO2 içeriği yaklaşık 400 ppm'dir (milyonda bir parça). İnsanlar soludukları havaya yaklaşık 40.000 ppm (= %4) CO2 yaydığından (dakikada 8-10 litre), havadaki CO2 yüzdesi birden fazla kişinin bulunduğu kapalı bir odada sürekli olarak artar. 1.200 ppm'ye ulaşırsa, odadaki havanın %2'si daha önce bir kez solunmuş olacaktır - bu nedenle, her 50. nefes, solunan havadan oluşacaktır. Bu nedenle, 1.000-2.000 ppm'lik bir CO2 seviyesi Federal Çevre Ajansı tarafından endişe verici olarak değerlendirilmektedir; özellikle okul sınıflarında, ortalama 1.000 ppm'yi aşmamalıdır [5].

Nefes verdiğimiz havada küçük damlacıklar (aerosoller) da bulunur; bunlar havada uzun süre kalabilen virüsler içerebilir (2,7 saatlik bir virüs bulaşıcı aktivite yarı ömründe 10 cm/saat hızla düşer). Bu nedenle havadaki nemin artması, virüslerin odada yayılıyor olabileceğinin bir risk göstergesidir.

Alanın havanın geçmesine izin verecek şekilde kısa bir süre havalandırılması, CO2 seviyesini ve nemi hızlı ve etkili bir şekilde azaltabilir. Enfeksiyona karşı koruma sağlamak için sınıflarda her 20 dakikada bir havalandırma yapılması önerilir [6].

2a. Öncelikle, sınıfın havalandırılması için kullanılması gereken nem ve hava kalitesi için eşik değerlerini belirleyin. Bunu yapmak için, örneğin 20 dakika sonra sınıfta ölçülen değerleri belirleyebilirsiniz. Hedef değerler, havalandırma yoluyla elde edebileceğiniz nem ve hava kalitesi seviyeleridir.

IoT sunucusunun panosundaki aşağıdaki yardımcı programı kullanarak ölçülen değerler, bağlı nem ve hava kalitesine ilişkin eğriyi görüntüleyebilirsiniz:



IoT_MQTT_Indoor_Air_Quality.ft

Eşik değerlerini kullanarak bir “havalandırma endeksi” türetin (0-100 ile: 0 = havalandırmadan sonra ideal durum, 100 = acilen havalandırmaya ihtiyaç duyuluyor), hesaplamaya bağlı nem ve hava kalitesini 1:3 ağırlıkla dahil edin.

2b. Kırmızı LED'in her 0,5 saniyede bir yanıp sönmeye neden olan bir alt program programlayın. Kırmızı yanıp sönen ışık 100 havalandırma indeksinde etkinleştirilmelidir.

2c. İkinci LED'i (yeşil kapaklı) O7 ve O8'e bağlayın (polariteye dikkat edin) ve havalandırma endeksiniz "0" a ulaştığında etkinleştirin. Sadece bir pencere, iki pencere, ... tüm pencere ve kapı(lar) açıkken havalandırma süresini karşılaştırın.

3. Gürültü seviyesi

Çevremizdeki gürültü seviyesi de refahımızı etkileyebilir. Ses seviyesi, ses basınç seviyesi olarak ölçülür ve desibel (dB) birimi kullanılarak belirtilir. 0 dB, insan işitsel eşiğine karşılık gelir (20 mikropaskal civarında basınç dalgalanmaları); 120 dB genellikle insan dinleyiciler için ağrı eşiğidir.

(Algılanan) ses seviyesi desibel seviyesiyle logaritmik olarak artar: Ses basınç seviyesi her 10 dB arttığında yaklaşık olarak iki katına çıkar. Ses basınç seviyesi ise, gürültü kaynağına olan mesafeye bağlıdır: Mesafenin üç katına çıkarılması sesi yarıya indirir (ses basınç seviyesinde 10 dB azalma).

Karşılaştırmalar, belirli desibel seviyelerinin ses seviyelerini hissetmemize yardımcı olabilir: Tik tak eden bir çalar saatin sesi (1 metre mesafede) yaklaşık 30 dB, normal bir konuşmanın sesi yaklaşık 60 dB, yoldan geçen bir otobüsün sesi yaklaşık 80 dB, havalı bir matkap 100 dB ve bir jet uçağının sesi 120 dB'dir.

Yüksek hacim seviyelerinin önemli olumsuz etkileri olabilir:

- İnsanlar sadece 40 dB'de bile konsantre olmada sorun yaşayabilirler
- 60 dB'nin üzerindeki ses seviyeleri uzun süreli maruziyetin ardından ısıtma hasarına neden olabilir
- Kardiyovasküler hastalık riski 65+ dB'nin üzerindeki sürekli düzeylerde artar
- 120 dB'nin üzerinde ise kısa bir süre sonra işitme kaybı meydana gelebilir.

3a. Programlama görevi 2'deki (veya deneysel görev 1'deki) programınızın görüntüsüne ses seviyesini (dB cinsinden) gösterecek bir ölçek ekleyin.

3b. Kameradaki mikrofon, alınan ses seviyesini dB olarak gösterecektir. Ses ölçümünü ve gösterimini Blockly programınıza ekleyin.

3c. Kırmızı LED'in yanıp sönmesi gereken bir eşik değeri tanımlayın. Programınızı buna göre genişletin.

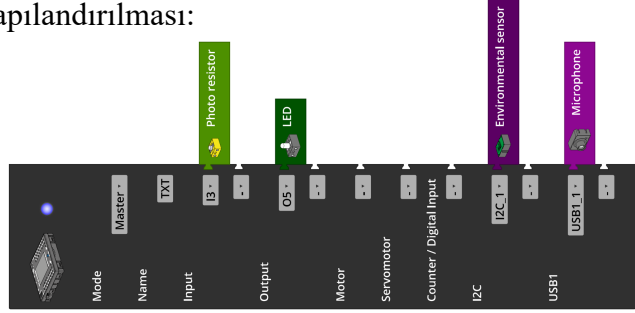


IoT Model İçin Verilen Çözümler

Çözüm sayfası - İç mekan iklimi (Görev -2)

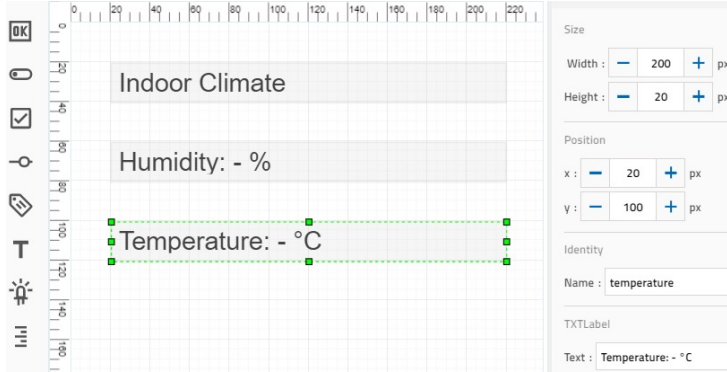
Programlama görevleri

Sensör ve aktüatörlerin yapılandırılması:



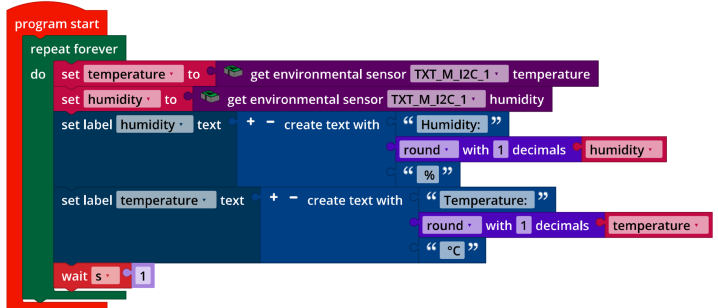
1. İç mekan iklim değerlerinin ölçülmesi

1a. TXT görüntüsünün yapılandırılması:



TXT'de görüntülemeyi yapılandırma

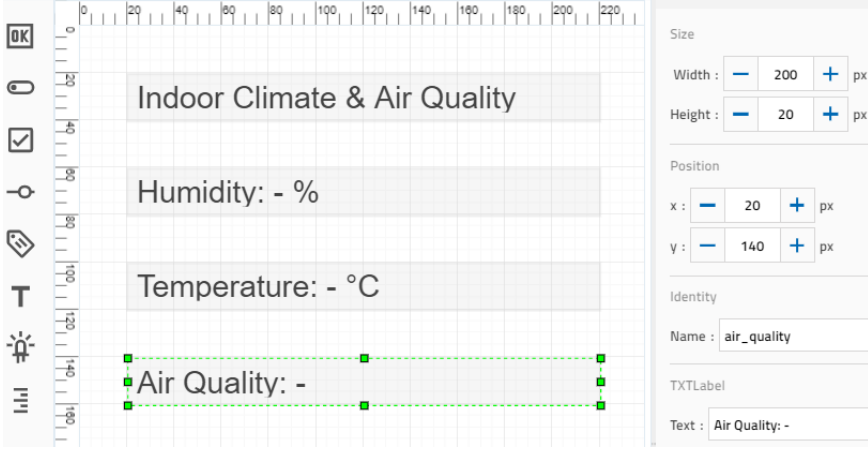
1b. Program (örnek):



IoT_Indoor_Climate.ft

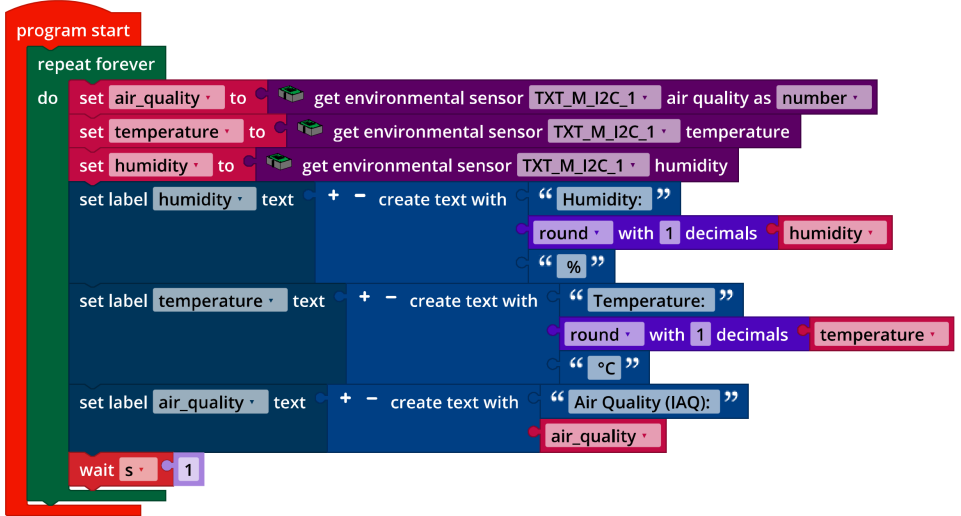
2. Hava kalitesinin ölçülmesi

2a. TXT görüntüsünün yapılandırılması:



TXT'deki ekranı genişletme

2b. Program (örnek):



IoT_Indoor_Air_Quality.ft

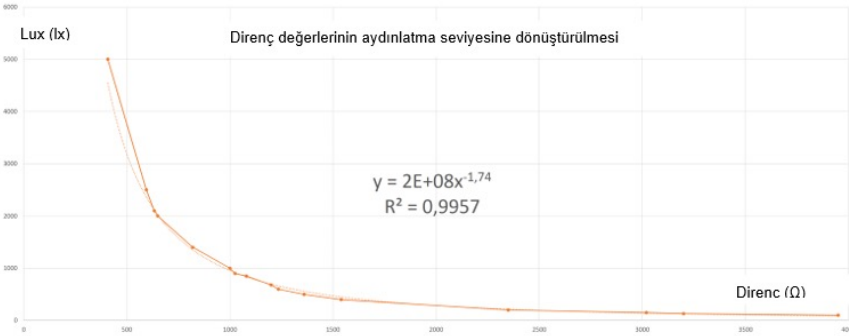
Deneyisel görevler

1. Parlaklık

Ölçülen değerler (direnc/ölçülen aydınlatma düzeyi) bir elektronik tablolama programı kullanılarak bir grafikte gösterilmelidir (x: direnc, y: aydınlatma düzeyi).

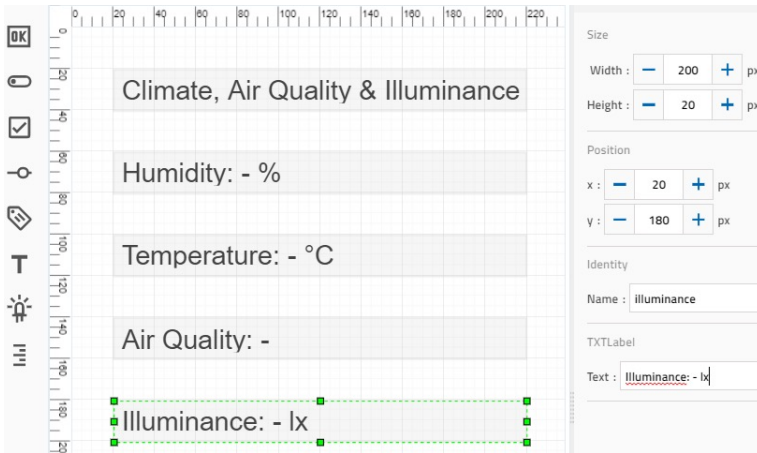
Direnç değeri r , karakteristik eğriden türetilen bir denklem kullanılarak yaklaşık olarak aydınlatma seviyesi i 'ye dönüştürülebilir. Örneğin Excel'de eklenen bir eğilim çizgisi için gösterilen denklem aşağıdaki gibi olabilir (gün ışığı ölçümlerine dayanarak):

$$i = 2 \cdot 10^8 \cdot r^{-1.74}$$



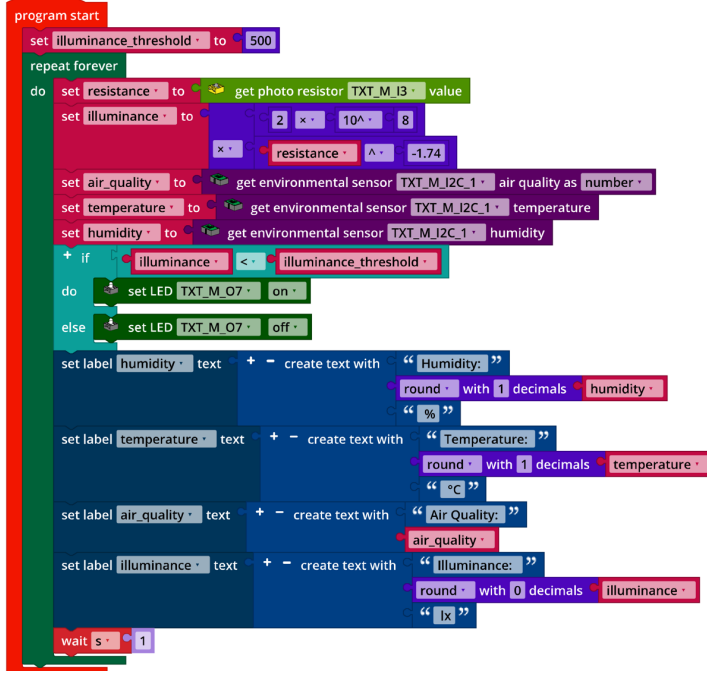
Çalışma istasyonu yönetmeliğine göre bir çalışma istasyonunun en az 500 lx aydınlatma seviyesine sahip olması gerekir; bu aynı zamanda bir sınıf ortamındaki çalışma istasyonu için de iyi bir asgari gerekliliktir.

TXT görüntüsünün yapılandırılması:



TXT'deki görüntü çıktısını genişletme

Program (örnek):



IoT_Illuminance.ft

2. CO2 trafik ışıkları

2a. “Havalandırma endeksi” L , 100’e kadar bir değere sahip olmalı, nem ve iç hava kalitesi değerlerinin her biri istenen hedef değere ulaştığında sıfıra eşit olmalı ve nem ve iç hava kalitesi değerleri “Şimdi havalandırın!” seviyesine ulaştığında 100 olmalıdır.

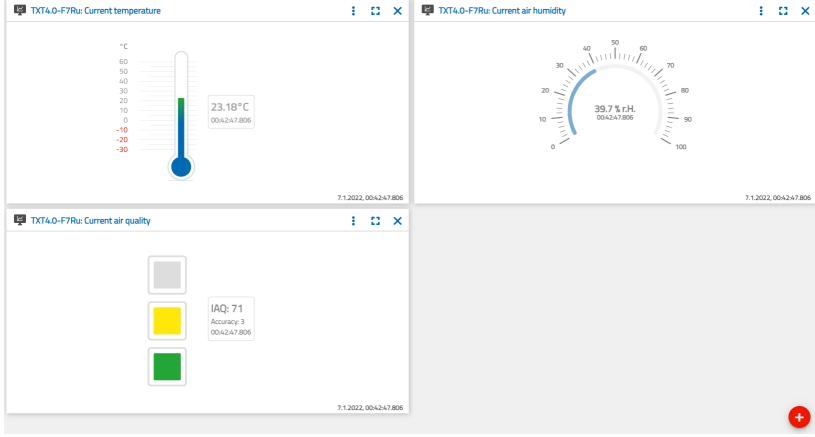
H_0 hedef değer ve H_1 bağıl nem için eşik değer (üst sınır) ve H ölçülen değer ise; ve ayrıca Q_0 hedef değer ise, Q_1 hava kalitesi için eşik değer (üst sınır) ve Q ölçülen IAQ değeri ise. O zaman, “havalandırma endeksi” L ’yi şu şekilde tanımlayabiliriz:

$$L = \left(\frac{H - H_0}{H_1 - H_0} + \frac{Q - Q_0}{Q_1 - Q_0} \cdot 3 \right) \cdot 25$$

Örnek hesaplama: Havalandırma sonrası bağıl nem %30 ve IAQ 30 (hedef değerler) olması gerekiyorsa ve eşik değerleri (örneğin sınıfta havalandırma olmadan 20 dakika sonra değerler) %55 ve %100 olarak tanımlanırsa, o zaman:

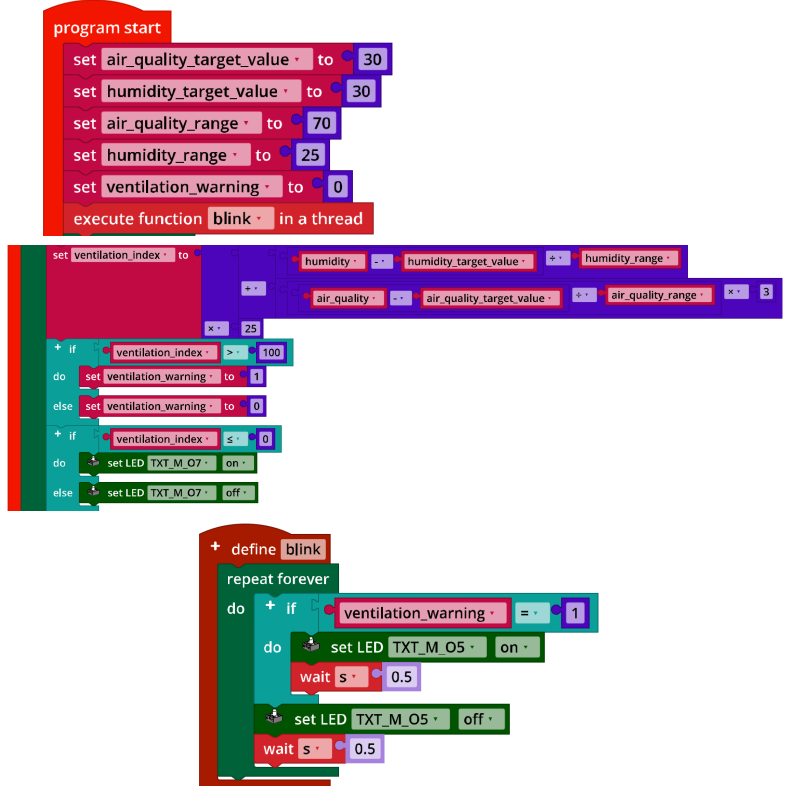
$$L = \left(\frac{H - 30}{55 - 30} + \frac{Q - 30}{100 - 30} \cdot 3 \right) \cdot 25$$

Ölçülen değerleri (sıcaklık, bağıl nem ve hava kalitesi) görüntülemek için IoT panosunun yapılandırılması:



IoT panosunun yapılandırması

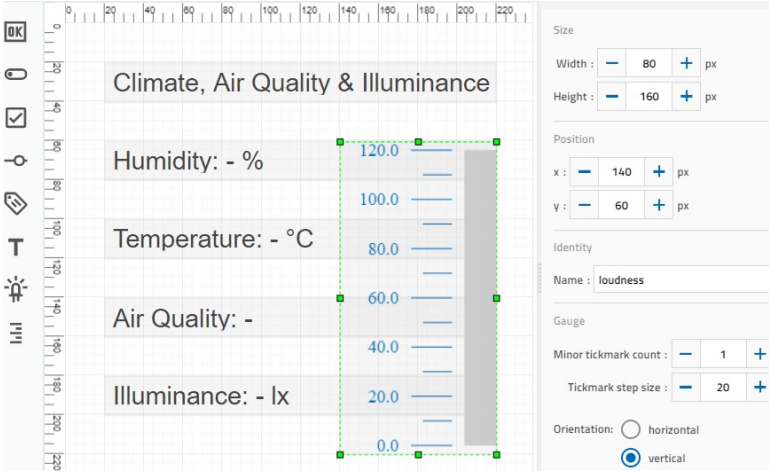
2b/c. Program özetleri (örnek):



IoT_CO2_Signal_Light

3. Gürültü seviyesi

3a. TXT görüntüsünün yapılandırılması:



TXT görüntüsünü bir ses ölçüğü ekleyerek genişletme

3c. Örneğin ses seviyesi için sınır değer olarak 70 dB seçilebilir.

```

program start
  set luminance_threshold to 500
  set loudness_threshold to 70
  set loudness_warning to 0
  execute function blink in a thread

  repeat forever
    do
      set loudness to microphone TXT_M_USB1_1 volume
      set resistance to get photo resistor TXT_M_I3 value
      set illuminance to
        K = resistance A = 1.74
        K * A
      set air_quality to get environmental sensor TXT_M_I2C1_1 air quality as number
      set temperature to get environmental sensor TXT_M_I2C1_1 temperature
      set humidity to get environmental sensor TXT_M_I2C1_1 humidity
      if illuminance < luminance_threshold
        do
          set LED TXT_M_O7 on
        else
          set LED TXT_M_O7 off
      if loudness > loudness_threshold
        do
          set loudness_warning to 1
        else
          set loudness_warning to 0

      set label humidity text + create text with "Humidity: "
      round with 1 decimals humidity
      set label temperature text + create text with "Temperature: "
      round with 1 decimals temperature
      set label air_quality text + create text with "Air Quality: "
      set label illuminance text + create text with "Illuminance: "
      round with 0 decimals illuminance
      set gauge loudness value loudness
      wait 5 1

    define blink
      repeat forever
        do
          if loudness_warning = 1
            set LED TXT_M_O5 on
            wait 5 0.5
            set LED TXT_M_O5 off
            wait 5 0.5
          end
        end
      end
  end

```

IoT_Loudness.ft



IoT Model İçin Öğrenme Hedefleri

Görev 3: Alarm sistemi

Sensör istasyonu, farklı tetikleyicilere (gürültü, hareket) tepki veren ve bir kamera görüntüsünü bir web sunucusuna ileten gözetim kamerasına sahip bir alarm sistemine dönüştürülür. Kamera, IoT panosundan kontrol edilebilir.

Konular

- Bir kameranın ses ve uzaktan kontrolü (“pan-tilt kontrolü”)
- Hareket ve gürültü aktivasyonu
- Bir IoT sisteminin uzaktan kontrolü için bir web sunucusunun kullanımı

Öğrenme hedefleri

- Enkoder motorları aracılığıyla hassas kontrol
- Fonksiyonları kullanarak bir programın sürdürülebilir yapılandırılması
- MQTT protokolünü anlamak

Gerekli malzemeler

- Program geliştirme için PC, lokal veya web arayüzü üzerinden.
- Programı TXT4.0'a iletmek için USB kablosu veya BLE veya WiFi bağlantısı.
- fischertechnik “Voice Control“ app
- Yardımcı program “IoT_Test_Dashboard_Control.ft”
- Fischertechnik bulutundaki hesap (www.fischertechnik-cloud.com)

Görev 3: Alarm sistemi

Artık sensör istasyonunuzu bir alarm ve gözetleme istasyonuna dönüştüreceksiniz: Seslere ve hareketlere tepki vermeli ve IoT panosunu kullanarak odayı izlemenize olanak sağlamalıdır.

Model Kurulumu

IoT modeli üzerinde herhangi bir değişiklik yapmadan kullanabilirsiniz.

Programı test etmeden önce, kablolanmanın kamerayı geniş açıda sola ve sağa çevirmeye yetecek kadar boşluk sağladığından emin olmalısınız.

Programlama görevleri

1. Ev pozisyonu

Alarm sisteminizdeki kameranın önce ana konuma getirilmesi gerekir. Bunu yapmak için, kamerayı ilgili tahrik motoruna göre yatay ve dikey olarak saat yönünün tersine çevirmeniz gerekir; ta ki ilgili durdurma konumu sensörleri etkinleştirilene kadar, böylece kameranın tam konumunu bilirsiniz. Buradan, kamerayı istediğiniz ana konuma getirebilirsiniz.

1a. Kamerayı yatay olarak öne hizalamak için kodlayıcı motorun kaç darbesi ile kamera uç konumundan dikey olarak yukarı doğru döndürülmelidir? Önce dişli oranını kullanarak değeri belirleyin.

Daha sonra uygun bir Blockly programı kullanarak kamerayı önce son konuma çevirip, sonra yatay olarak hizalayarak sonucu deneysel olarak kontrol edin.

Not: Hatırlatma: Enkoder motoru eksen dönüşü başına 63,9 darbe çıkışı verir.

1b. Kodlayıcı motorun kaç darbesi ile kamera uç konumundan yatay olarak saat yönünde döndürülmeli ki "düz ileri" hizalansın? Değeri önce dişli oranını kullanarak belirleyin. Daha sonra, kamerayı önce uç konuma döndüren ve sonra öne hizalayan uygun bir Blockly programı kullanarak sonucu deneysel olarak kontrol edin.

Not: Döner halkanın 58 dişi vardır.

1c. Her iki programı, kamerayı her iki boyutta hizalayan bir fonksiyona birleştirin. İki hareketi paralel olarak nasıl yürütebilirsiniz?

2. Kamera gözetimi

Görev 1'de, hava durumu istasyonunu her saniye web kamerasından gösterge paneline bir görüntü gönderecek şekilde programladınız. Şimdi odayı aynı şekilde kamerayla izleyebilirsiniz.

Programlama görevi 1'den programınızı, panoya görüntü iletimi ekleyerek genişletin. Panoda görüntülenecek her saniye iletebileceğiniz maksimum görüntü sayısını belirleyin.

3. Gürültü aktivasyonu

Güvenlik personeli, hiçbir şeyin olmadığı ekranları sürekli izlemek zorunda kalırsa yorgun düşebilir. Bu nedenle, görüntü iletimi yalnızca mikrofon alışılmadık bir gürültü kaydettiğinde başlamalıdır.

3a. Öncelikle, basit bir Blockly programı kullanarak odadaki "normal" gürültü seviyesini test edin. Değerleri TXT'de görüntüleyin. Ardından, görüntü iletiminin başlaması gereken uygun bir eşik değeri ayarlayın.

Not: Mikrofon için Blockly komutu ölçülen ses seviyesini size desibel (dB) cinsinden gösterecektir.

3b. Tanımladığınız gürültü seviyesi eşik değeri aşıldığında her saniye kamera görüntüsünü iletmeye başlayan bir Blockly programı yazın. Eğer her şey bir dakikadan fazla sessiz kalırsa, görüntü iletimi durmalıdır.

4. Hareket algılama

Ayrıca kamera gözetimini odadaki bir hareketle etkinleştirilecek şekilde ayarlayabilirsiniz. Bu amaçla kamera hareket algılamayı yapılandırın.

4a. Blockly programınızı, hareket algılandığında görüntü iletimi de etkinleştirilecek şekilde programlama görevi 3'ten genişletin.

4b. Kamera aktif olduğunda kırmızı LED'in de yanıp sönmesi gerekir.

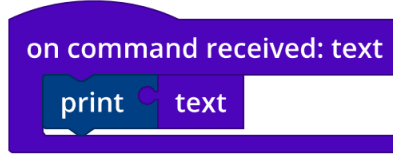
Deneyisel görevler

1. Ses kontrolü

Kamera artık odanın daha büyük bir bölümünü izleyebilmesi için sesli komutlar kullanılarak kontrol edilmelidir. Bunu yapmak için, Apple uygulama mağazasından (iOS için) veya Google Play mağazasından (Android için) “Voice Control” uygulamasını indirin ve TXT 4.0'a bağlayın.

WiFi üzerinden bağlantı: TXT 4.0 Kontrol Cihazı ve cihaz (akıllı telefon veya tablet) aynı WiFi yönlendiricisine bağlı olmalıdır. Yönlendirici ayrıca cihazlar arasında iletişime izin vermelidir. Uygulamanın bağlanması gereken TXT 4.0'ın IP adresi daha sonra "Bilgi" / "WiFi" altındaki dokunmatik ekran menüsü üzerinden sorgulanabilir.

- WiFi AP üzerinden bağlantı: “WiFi” yerine, “Ayarlar” / “Ağ” altında TXT 4.0'da “Erişim Noktası” seçeneği etkinleştirilebilir. Ardından, akıllı telefon doğrudan denetleyiciye bağlanabilir. WiFi bağlantısı için gereken WPA2 anahtarı, “Erişim Noktası” altında TXT menüsünde mevcuttur (orada da değiştirilebilir ve devre dışı bırakılabilir).
- Uygulamayı Kontrol Ünitesine bağladığınızda, sesli komutlar Kontrol Ünitesine metin biçiminde iletilir ve aşağıdaki olay fonksiyonunu kullanarak bunları analiz edebilirsiniz:



1a. Sağa, sola, yukarı ve aşağı dönüş için işlevler yazın. Ses tanıma hafif bir gecikmeyle tepki verdiği için, her erişildiğinde kamerayı bir parametre olarak iletilen bir açıyla döndürmek mantıklıdır.

Aşağıdakileri göz önünde bulundurun: Kameranın çok fazla dönmesini nasıl önleyebilirsiniz?

1b. Kamera kontrolü için uygun ses komutlarını seçin. Alt görev 1a'daki kontrol fonksiyonlarını programlama görevi 4'teki programınıza ekleyin.

Şimdi, sesli komutlarınızı kullanarak kamerayı kontrol edin. Kameranın çok fazla dönmediğinden emin olun.

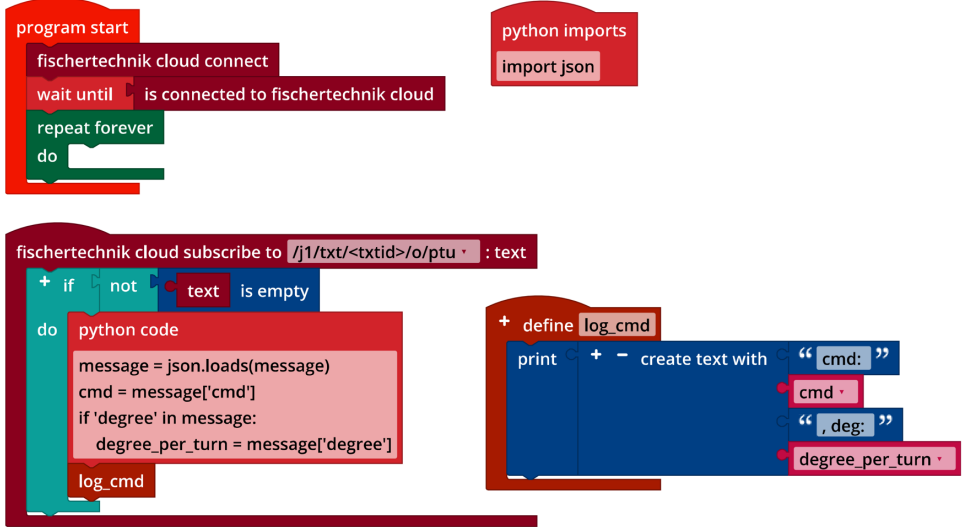
2. Bulut panosunu kullanarak kamerayı kontrol etme

Ayrıca fischertechnik bulutundaki IoT panosunu kullanarak kamerayı kontrol edebilirsiniz.

Bunu yapmak için IoT sunucusuna bağlandıktan sonra aşağıdaki fonksiyonu kullanarak TXT'yi “MQTT abonesi” olarak kaydetmeniz gerekir:

fischertechnik cloud subscribe to /j1/txt/<txtid>/o/ptu : text

2a. Daha sonra fare ile tıkladığınız komutu aşağıdaki test programı ile okuyabilirsiniz:



IoT_Test_Dashboard_Control.ft

2b. Deneysel görev 1'deki kontrol komutlarına ek olarak, kamerayı ana konuma taşımak için tasarlanmış bir komut ve motorları hemen durduran bir komut da var. Kontrol komutlarını buna göre ekleyin.

2c. Artık kamera ses kontrolünü, gösterge paneli üzerinden kontrolle değiştirebilirsiniz.

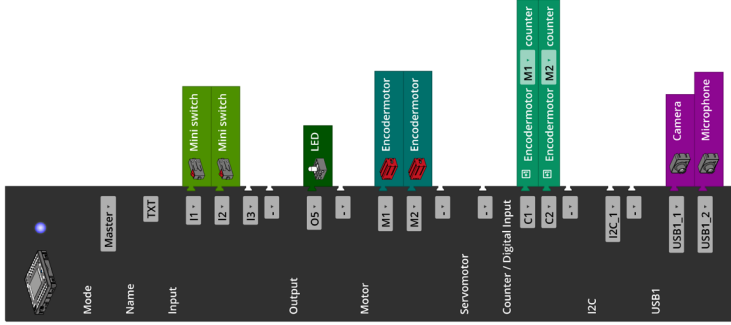


IoT Model İçin Verilen Çözümler

Çözüm sayfası – Alarm Sistemi (Görev -3)

Programlama görevleri

Sensör ve aktüatörlerin yapılandırılması:



Deneyisel görev 2'yi çözmek için “Ses Kontrolü” uygulaması (iOS veya Android için) gereklidir. Ses tanıma için uygulama internete bağlı olmalı ve Kontrol Ünitesine (Bluetooth veya WiFi üzerinden) bağlı olmalıdır.

1. Ev pozisyonu

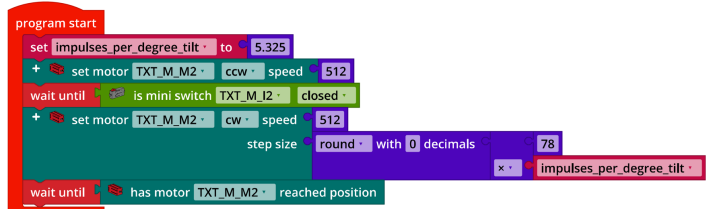
1a. Motor ekseninin her dönüşü, solucanın bir dönüşüne karşılık gelir ve solucanın her dönüşü Z30'u bir diş kadar döndürür. Bu nedenle, Z30'u 90° döndürmek için motor ekseninin 30/4 dönüşü gerekir. Bu, (eksenel dönüş başına 63,9 kodlayıcı darbesinde)

$$30 \cdot \frac{63.9}{4} = 479.25$$

Kodlayıcı darbeleri (veya açı derecesi başına 5.325 darbe).

Gerçekte, son konum anahtarı çeyrek turdan sonra tam olarak tetiklenmez; kameranın sadece 78° döndürülmesi gerekir. Bu, $78/90 \cdot 479.25 = 415.35$ kodlayıcı darbesine karşılık gelir.

Program (örnek):



IoT_Init_Camera_Tilt.ft

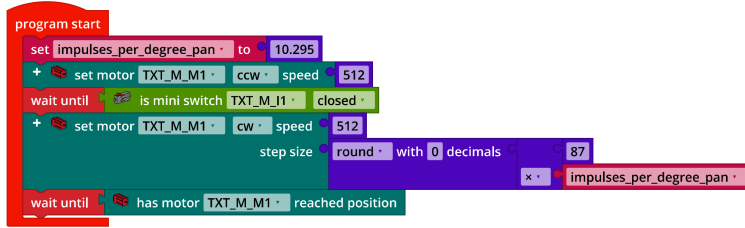
1b. Döner halka 58 dişe sahiptir; bu nedenle 90° dönüş, motor ekseninin 58/4 dönüşünü gerektirir. Bu,

$$58 \cdot \frac{63.9}{4} = 926.55$$

Kodlayıcı darbeleri (veya açıl derecesi başına 10.295 darbe).

Burada da son konum anahtarı sadece 90° dönüşten sonra tetiklenmez, yaklaşık 3°'de tetiklenir; kameranın sadece yaklaşık 87° döndürülmesi gerekir. Bu $87/90 \cdot 926.55 = 895.67$ kodlayıcı darbesine karşılık gelir.

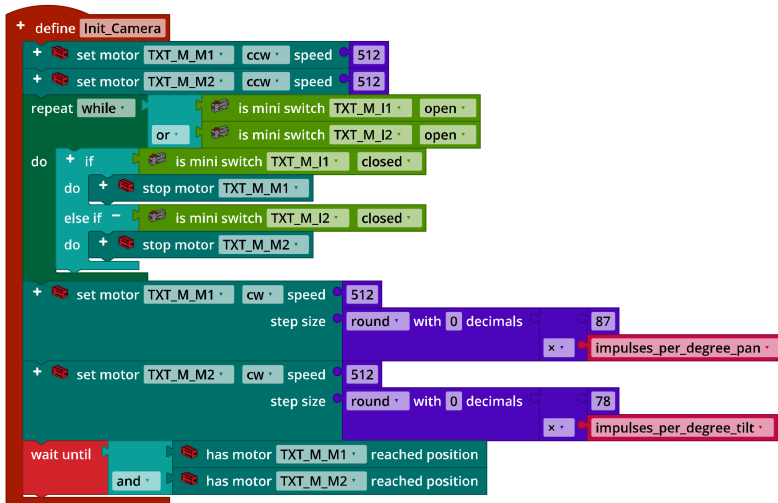
Program (örnek):



IoT_Init_Camera_Pan.ft

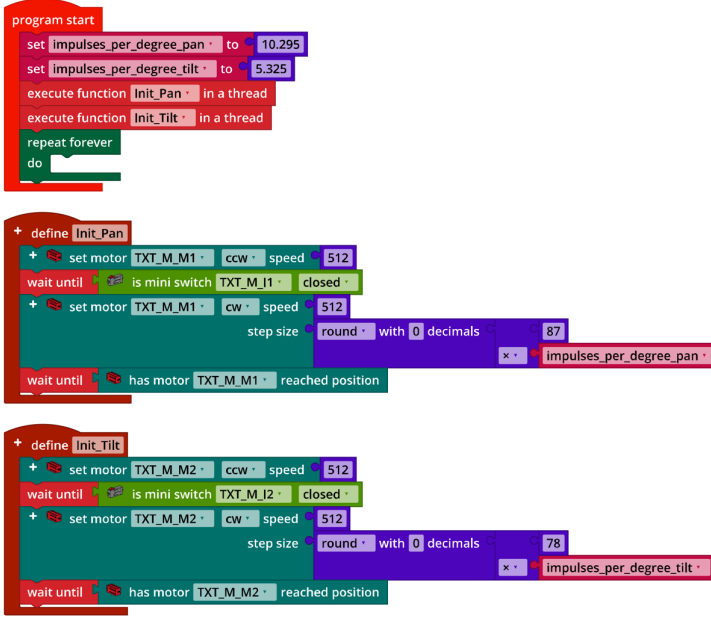
1c. Her iki program birleştirildiğinde, iki hareket paralel olarak gerçekleştirilebilir ve böylece başlatma süresi önemli ölçüde kısaltılabilir. Bunun için iki olası çözüm vardır: İki işlevi paralel iş parçacıkları olarak yürütmek veya bunları tek bir işleve entegre etmek.

Program çözüm yöntemi A – bir fonksiyon (örnek):



IoT_Init_Camera_A.ft

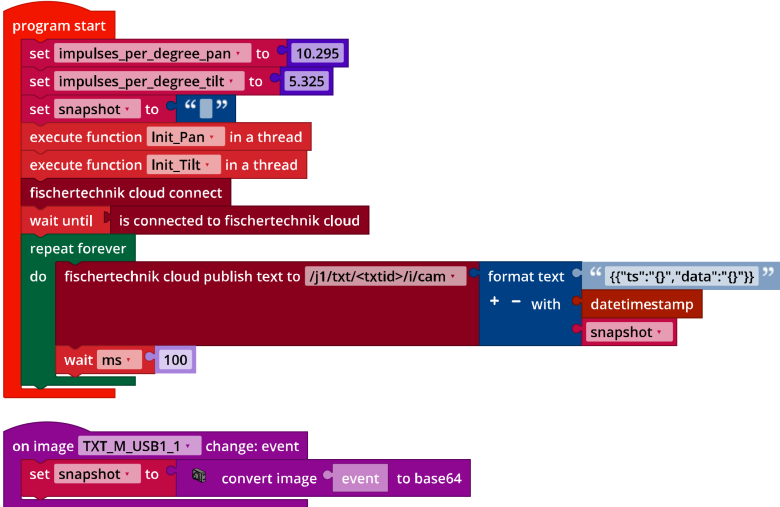
Program çözüm yöntemi B – paralel iş parçacıkları (örnek):



IoT_Init_Camera_B.ft

2. Kamera gözetimi

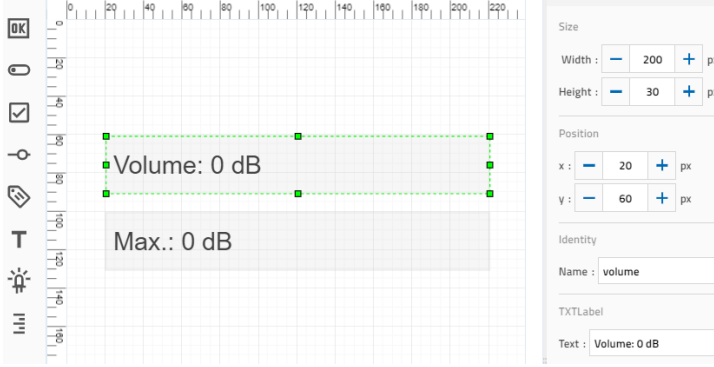
Program (örnek):



IoT_Surveillance_Camera.ft

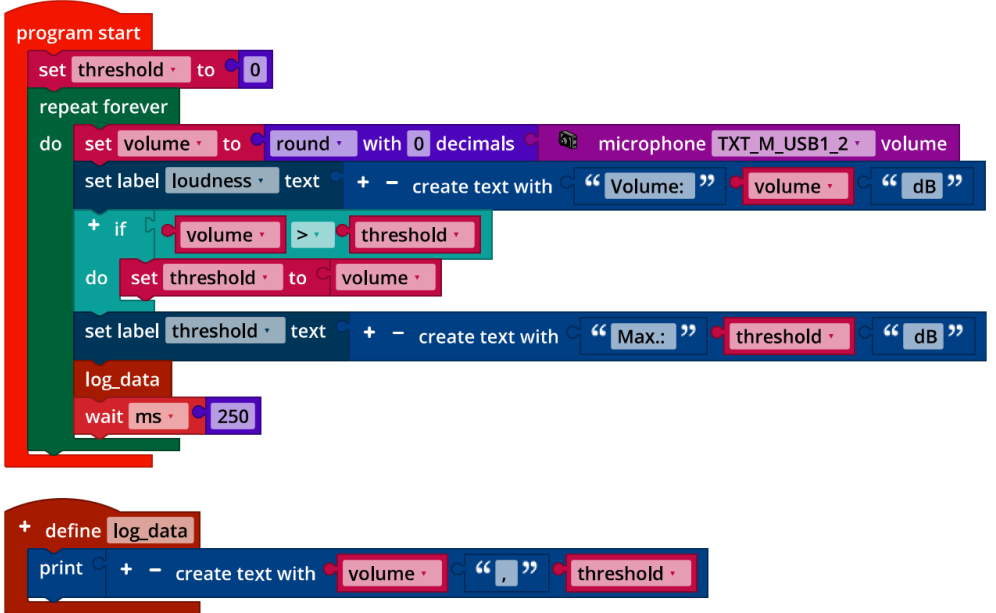
3. Gürültü aktivasyonu

3a. TXT ekranının yapılandırması (örnek):



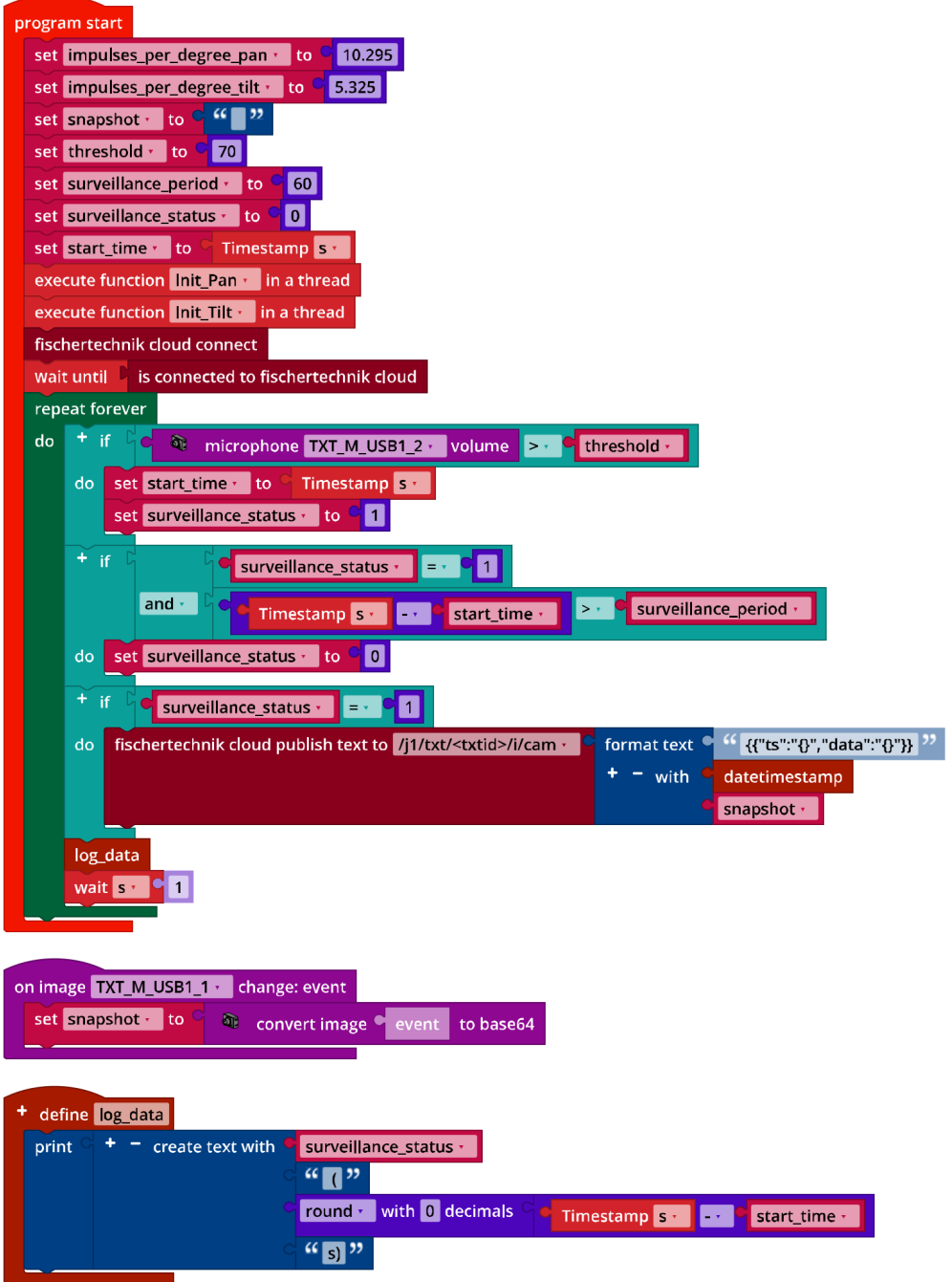
Görüntü yapılandırması

Program (örnek):



IoT_Microphone_Level.ft

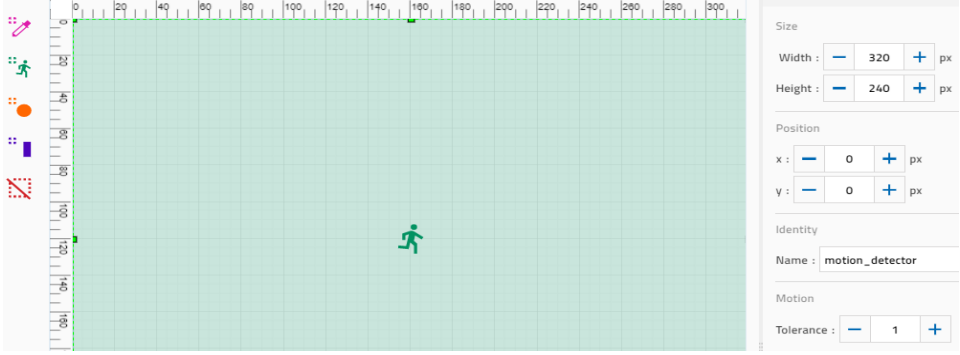
3b. Ses eşiği 70 dB olan program (örnek):



IoT_Surveillance_Camera_Noise_Detection.ft

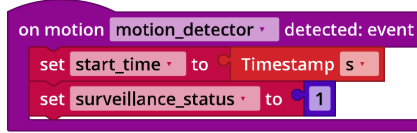
4. Hareket algılama

Hareket algılama, tüm kamera penceresi üzerinde tanımlanır. "Tolerans", denetçide algılama hassasiyetini 0 ila 1 değerine ayarlamak için kullanılabilir:



Hareket algılamayı yapılandırma

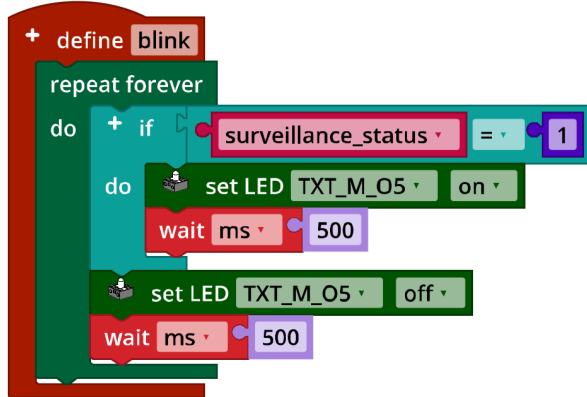
4a. Program özeti (örnek):



IoT_Surveillance_Camera_Motion_Detection.ft

Bir hareket algılanırsa, bu durum algılama zamanıyla birlikte bir bayrakta ("surveillance_status") kaydedilir.

4b. "Flash" fonksiyonu ana programın başlangıcında bir thread olarak başlatılır:

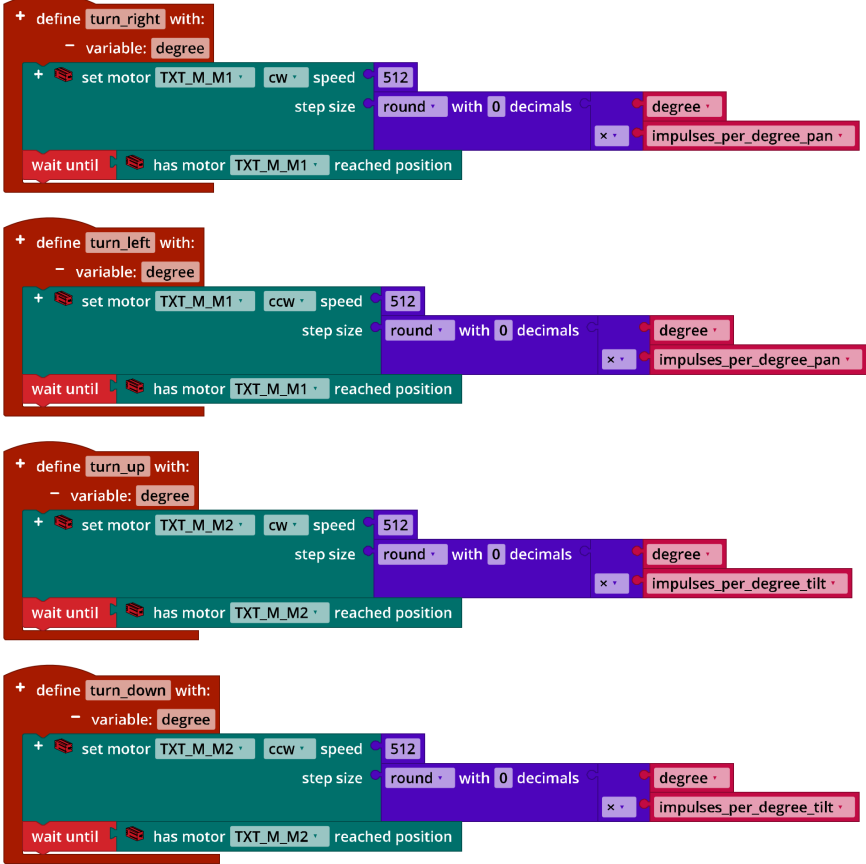


IoT_Surveillance_Camera_Motion_Detection.ft

Deneyisel görevler

1. Ses kontrolü

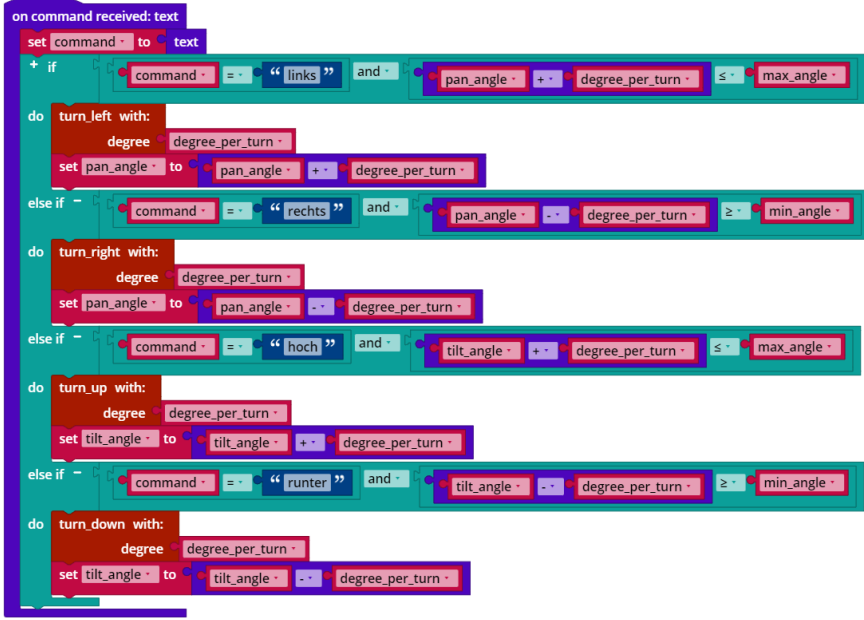
1a. Kontrol fonksiyonları (örnek):



IoT_Camera_Motion_Control.ft

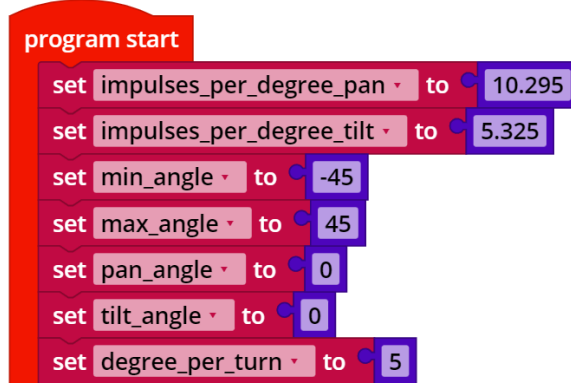
Kameranın çok fazla dönmesini engellemenin en kolay yolu, hem yatay hem de dikey yöndeki dönüş açısını toplamda $\pm 90^\circ$ ile sınırlamaktır. Ana konumu başlattıktan sonra her iki sapma açısını da 0° olarak ayarlarsanız, programı kontrol etmek kolaydır.

1b. Program özeti: Kamera kontrolü (örnek):



IoT_Surveillance_Camera_Voice_Control.ft

Aşağıdaki değişkenler ana programın başlangıcında başlatılır:



IoT_Surveillance_Camera_Voice_Control.ft

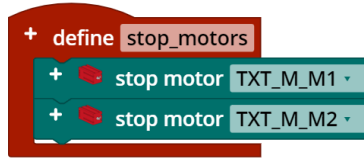
Gerekirse maksimum ve minimum sapma açıları tasarıma göre ayarlanmalıdır (kablolama aralığı).

2. Bulut panosunu kullanarak kamerayı kontrol etme

Gösterge panelindeki kontrol düğmeleri “stop”, “home”, “relmove_left”, “relmove_right”, relmove_up” ve “relmove_down” komutlarını iletir. “relmove” komutları deneysel görev 1'deki dört kontrol fonksiyonuna karşılık gelir. IoT sunucusu tarafından bir “derece” (2, 5, 10 veya 20°) değeri iletilirse, programda belirtilen dönüş açısı ayarlanmalıdır. “Home” komutu kamerayı ana konuma taşır; “stop” komutu için iki motoru durduran bir fonksiyon eklenmelidir.

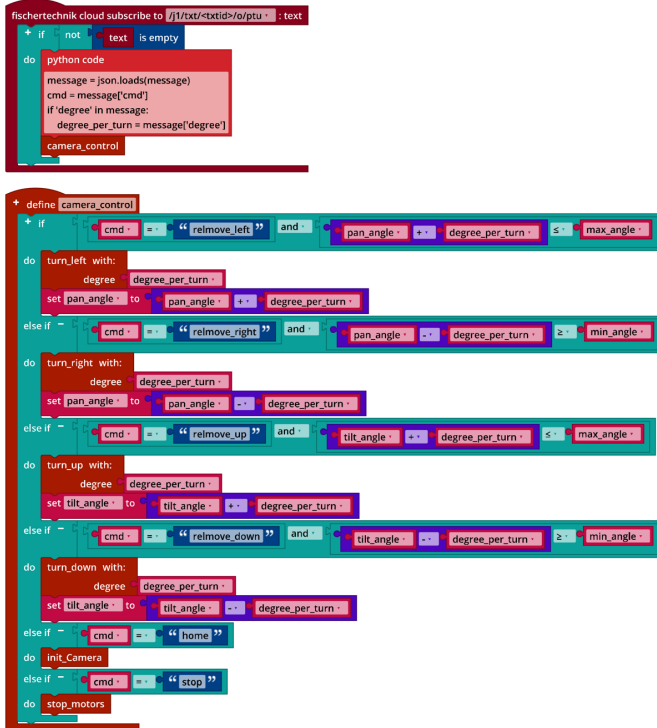
Kontrol komutunun sorgusu deneysel görev 1'in çözümünden alınabilir.

“stop_motors” fonksiyonu (örnek):



IoT_Surveillance_Camera_Dashboard_Control.ft

Program özeti: Kamera kontrolü (örnek):



IoT_Surveillance_Camera_Dashboard_Control.ft