

# INDINF\_7.2

---

Verfasser: **Leonhard Stransky, 4AHIT**

Datum: **26.09.2023**

## Projektbeschreibung

Ein Mikrocontroller mit Sensoren soll über eine elektrische Leitung oder einem Bussystem mit einem Roboter verbunden werden. Der Roboter soll die Daten des Mikrocontrollers auswerten und entsprechend reagieren.

## Projektteam:

- Julian Neuwirth
- Duchon Kevin
- Bakshi Pavel

## Umsetzung:

Wir haben uns für die Variante mit dem Bussystem zum Roboter entschieden. Hierbei wird ein Arduino Uno mit einem EasyCat Shield verbunden um Daten vom Mikrocontroller über ein Ethernetkabel an den Roboter zu senden.

Ein teil des Teams befasste sich mit der Implementierung von einem Distance Sensor.

Währenddessen habe ich die Online Dokumentationen von Bausano für das EasyCat Shield durchgelesen. In der Arduino IDE habe ich dann gleich die von der Website heruntergeladene Library eingebunden und ein Beispielprogramm am Arduino Uno ausprobiert.

Das EasyCat Shield verwendet ein Protokoll namens EtherCat um Datenpakete zu senden. Diese können im Code erstellt werden oder indem man mit dem Easy Configurator eigene Entries erstellt. Die standard firmware hat 32 + 32 bytes an Input und Output Entries.

Dort habe ich dann ein Input Entry erstellt, dass ein acknowledgement signal vom Roboter sein soll. (ok uint8\_t) und ein Output Entry, dass die Daten vom Distance Sensor beinhaltet. (distance float)

Diese Einstellungen musste man dann zum EEPROM des EasyCat Shields flashen. Dann wurde auch ein .h file erstellt was man in den Ordner des Arduino Projekts kopieren musste. Dieses File beinhaltet die Adressen der Entries.

Julian Neuwirth und ich haben dann ein Programm geschrieben, dass die Daten mittels dem EasyCat Shield an den Roboter sendet.

```
#include "INDINF.h"  
#include "EasyCAT.h"  
#include <SPI.h>
```

```
EasyCAT EASYCAT;
```

```
const int sharpPin = A0; // Der Sensor ist am analogen Pin A0 angeschlossen
float voltage;
float distance;

unsigned long PreviousMillis = 0;

void setup()
{
    Serial.begin(9600);

    Serial.print("\nEasyCAT - Generic EtherCAT slave\n");

    if (EASYCAT.Init() == true)
    {
        Serial.print("initialized");
    }
    else
    {
        Serial.print("initialization failed");
        pinMode(13, OUTPUT);
        while (1)
        {
            digitalWrite(13, LOW);
            delay(500);
            digitalWrite(13, HIGH);
            delay(500);
        }
    }

    PreviousMillis = millis();
}

void loop()
{
    EASYCAT.MainTask();

    int sensorValue = analogRead(sharpPin);
    voltage = sensorValue * (5.0 / 1023.0);

    distance = 27.86 * pow(voltage, -1.15);

    Serial.print("Entfernung: ");
    Serial.print(distance);
    Serial.println(" cm");

    // Fügen Sie die folgende Zeile hinzu, um die Variable distance in den EtherCAT-
    // Datenpuffer zu schreiben.
    EASYCAT.BufferIn.Float[0] = distance;
```

```
    delay(100);  
}
```

Jedoch leider hat der Code weder in der alten IDE noch in der neuen funktioniert.

Es war dann auch nicht mehr genug Zeit um das Problem zu lösen. Wir haben dann beim nächsten mal das Projekt an das andere Team übergeben. Dieses fand später raus, dass der code in einen custom mode gesetzt werden muss. Das stand in einem Example file der Library.

```
#define CUSTOM           // Custom mode
```

## Quellen:

[1] [www.bausano.net](https://www.bausano.net) 2023. *arduino-easycat* [online] Available at:  
<https://www.bausano.net/it/hardware/arduino-easycat.html> [Accessed 26 November 2023].