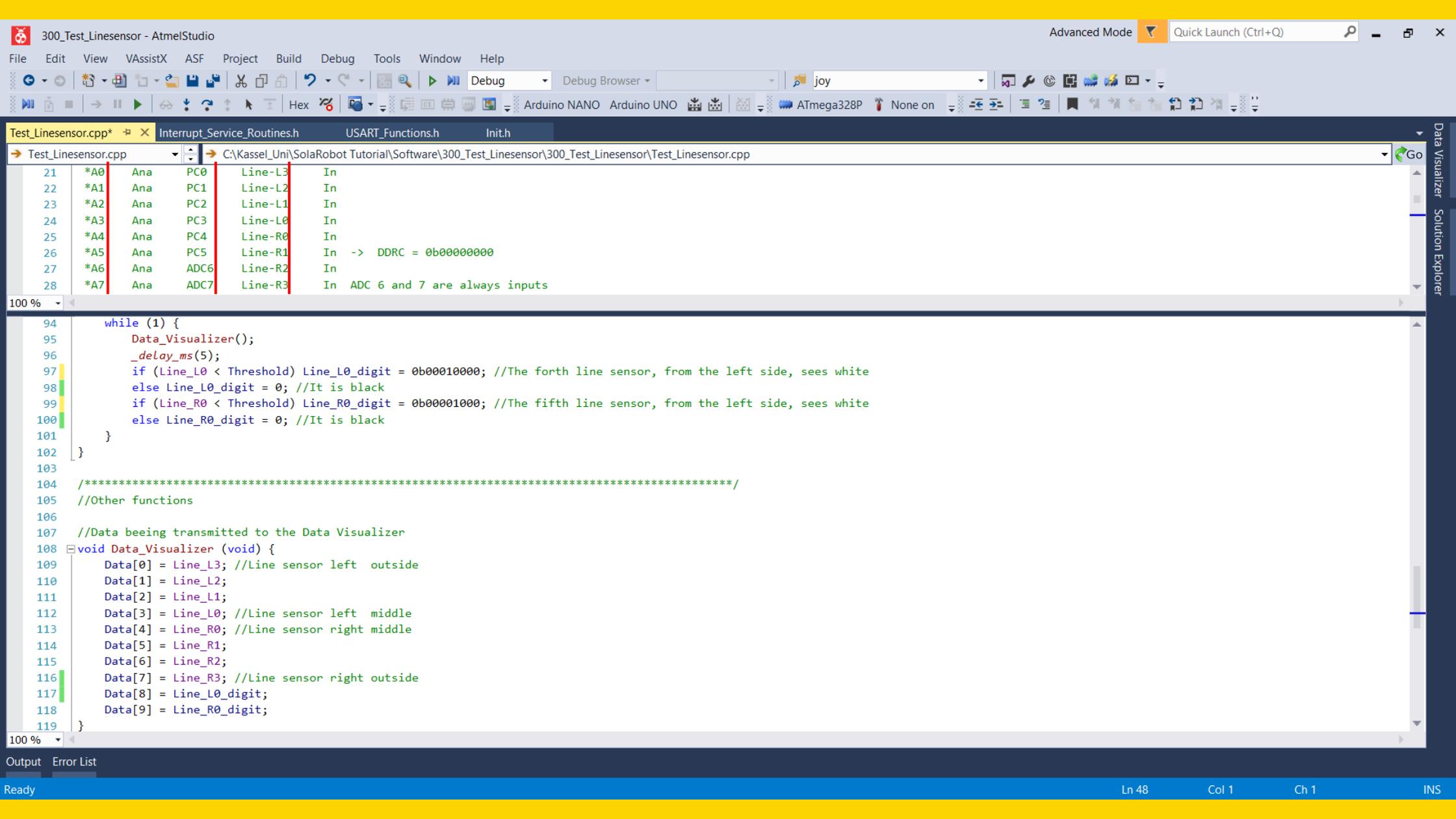


Montage und Programmierung
eines Roboters für
den Hessen SolarCup
Disziplin: SolaRobot
Teil 2.4: Line Sensor

Von Charlotte und Andreas

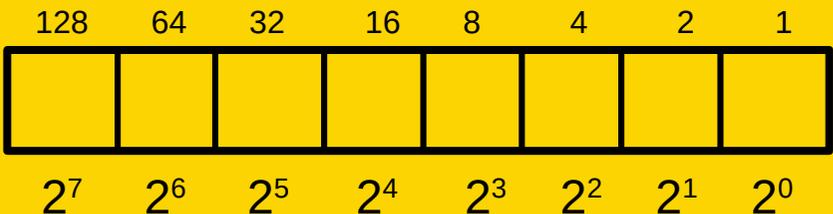


Linien-Sensor:

Liniensensor über weiß,
entsprechendes Bit High (= 1)
Über schwarz Bit Low (= 0)

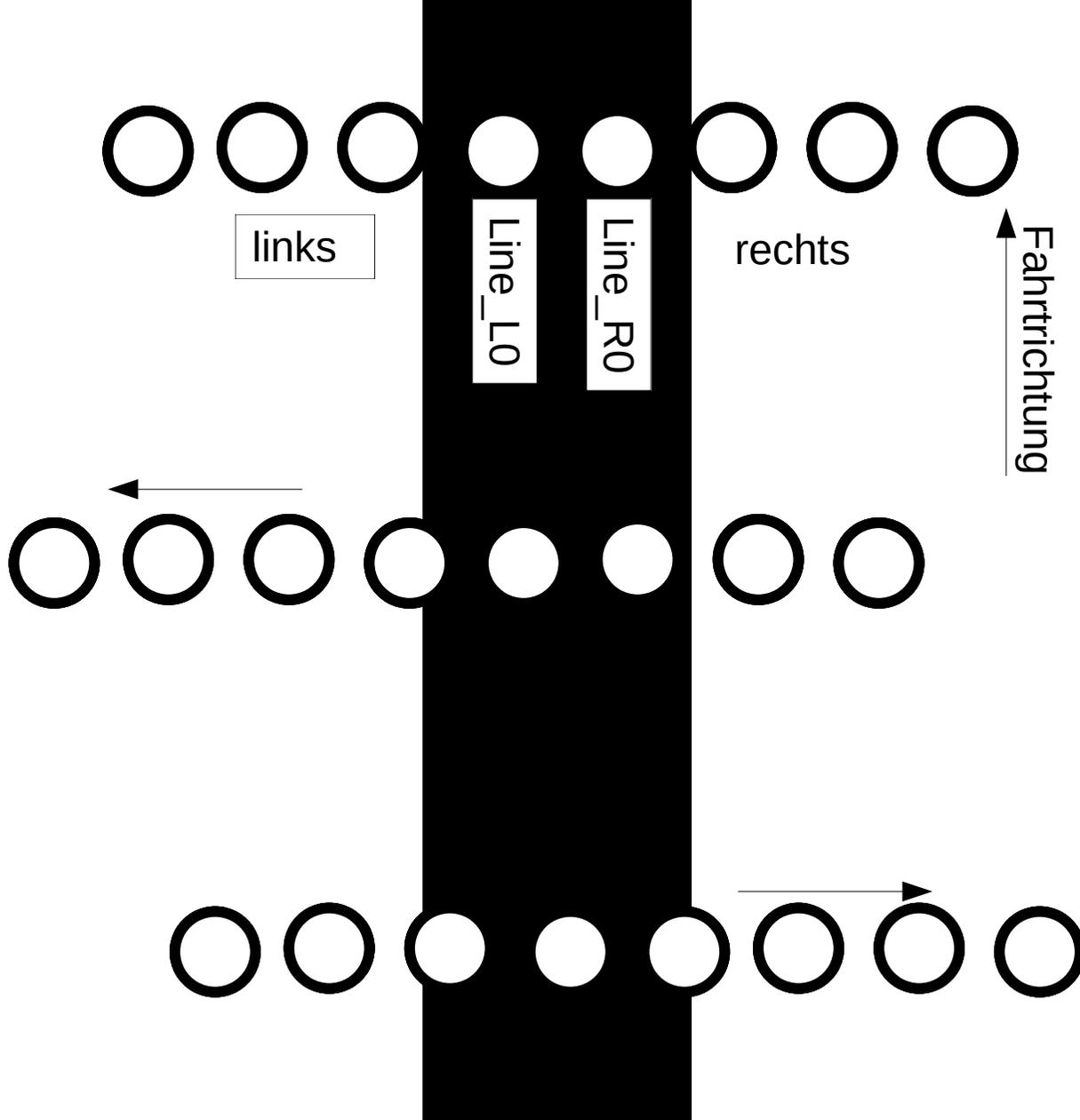
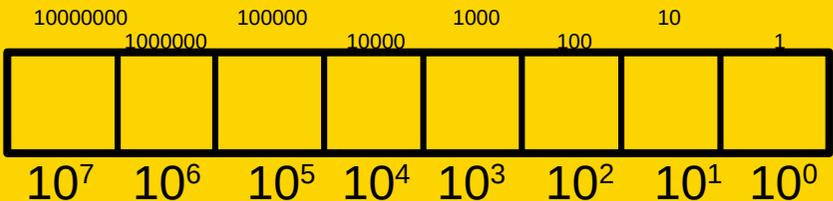
Rechter Sensor Bit 3
Linker Sensor Bit 4

Stellenwerte der Bits



Bit 3 hat den Stellenwert 8
Bit 4 hat den Stellenwert 16

Stellenwerte von Dezimalzahlen

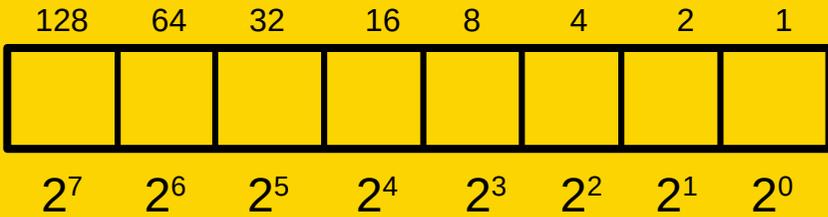


Linien-Sensor:

Liniensensor über weiß,
entsprechendes Bit High (= 1)
Über schwarz Bit Low (= 0)

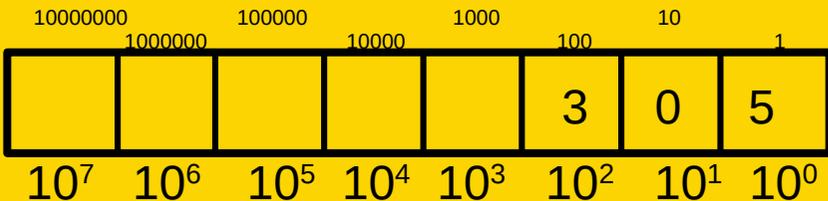
Rechter Sensor Bit 3
Linker Sensor Bit 4

Stellenwerte der Bits

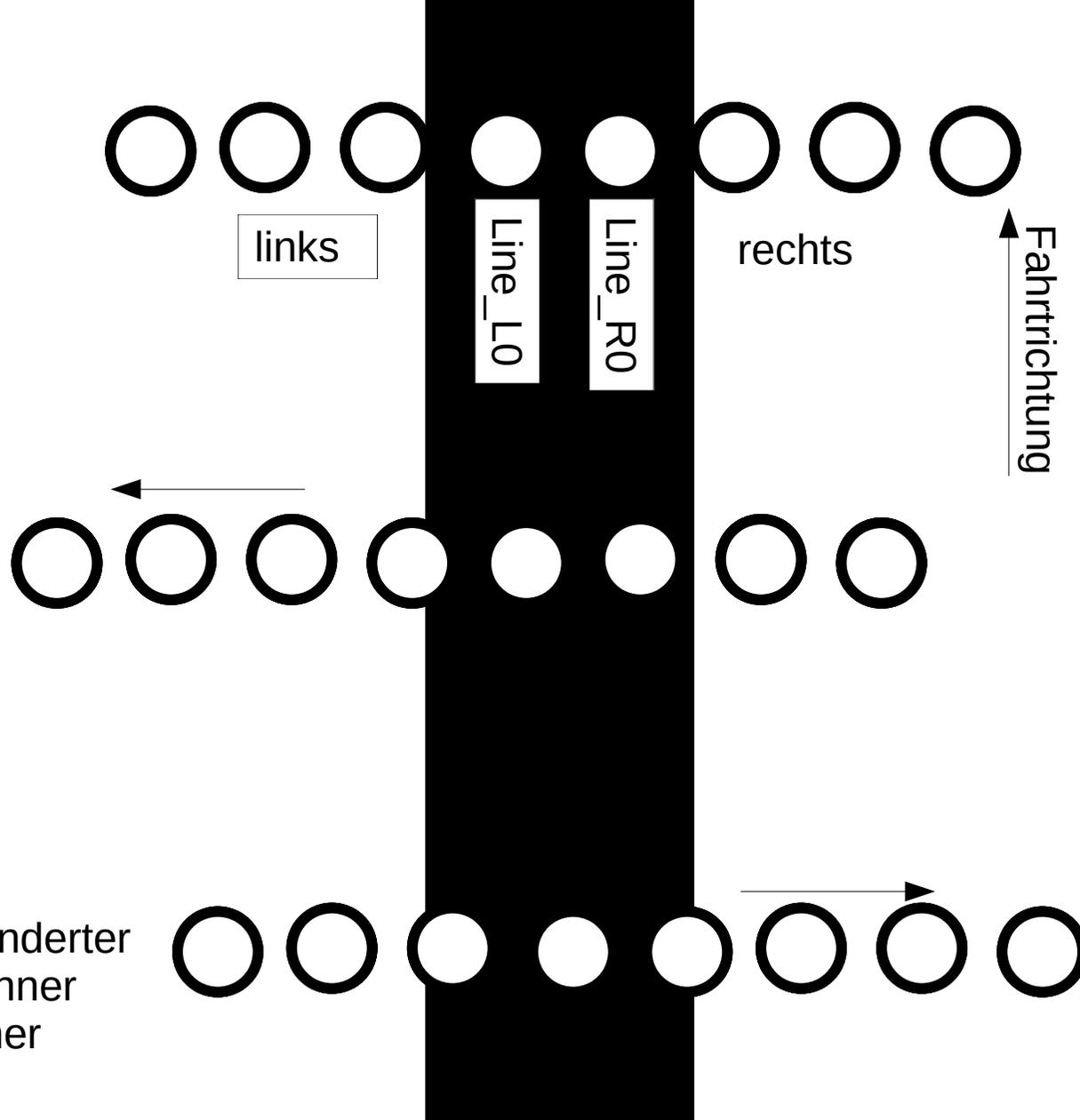


Bit 3 hat den Stellenwert 8
Bit 4 hat den Stellenwert 16

Stellenwerte von Dezimalzahlen



3 Hunderter
0 Zehner
5 Einer

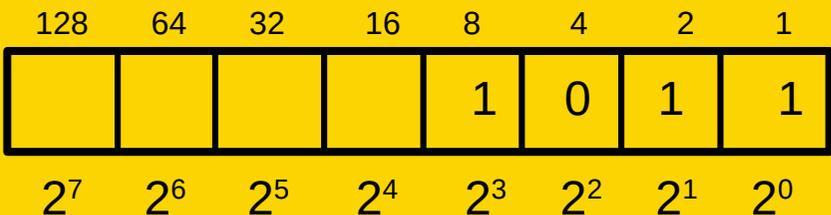


Linien-Sensor:

Liniensensor über weiß,
entsprechendes Bit High (= 1)
Über schwarz Bit Low (= 0)

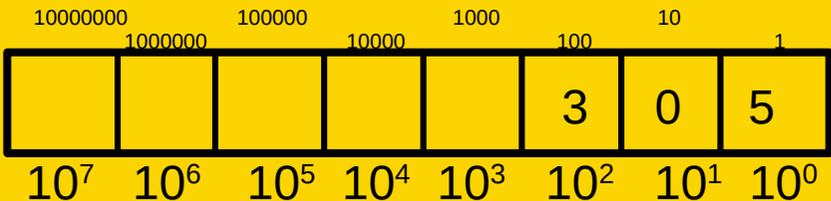
Rechter Sensor Bit 3
Linker Sensor Bit 4

Stellenwerte der Bits



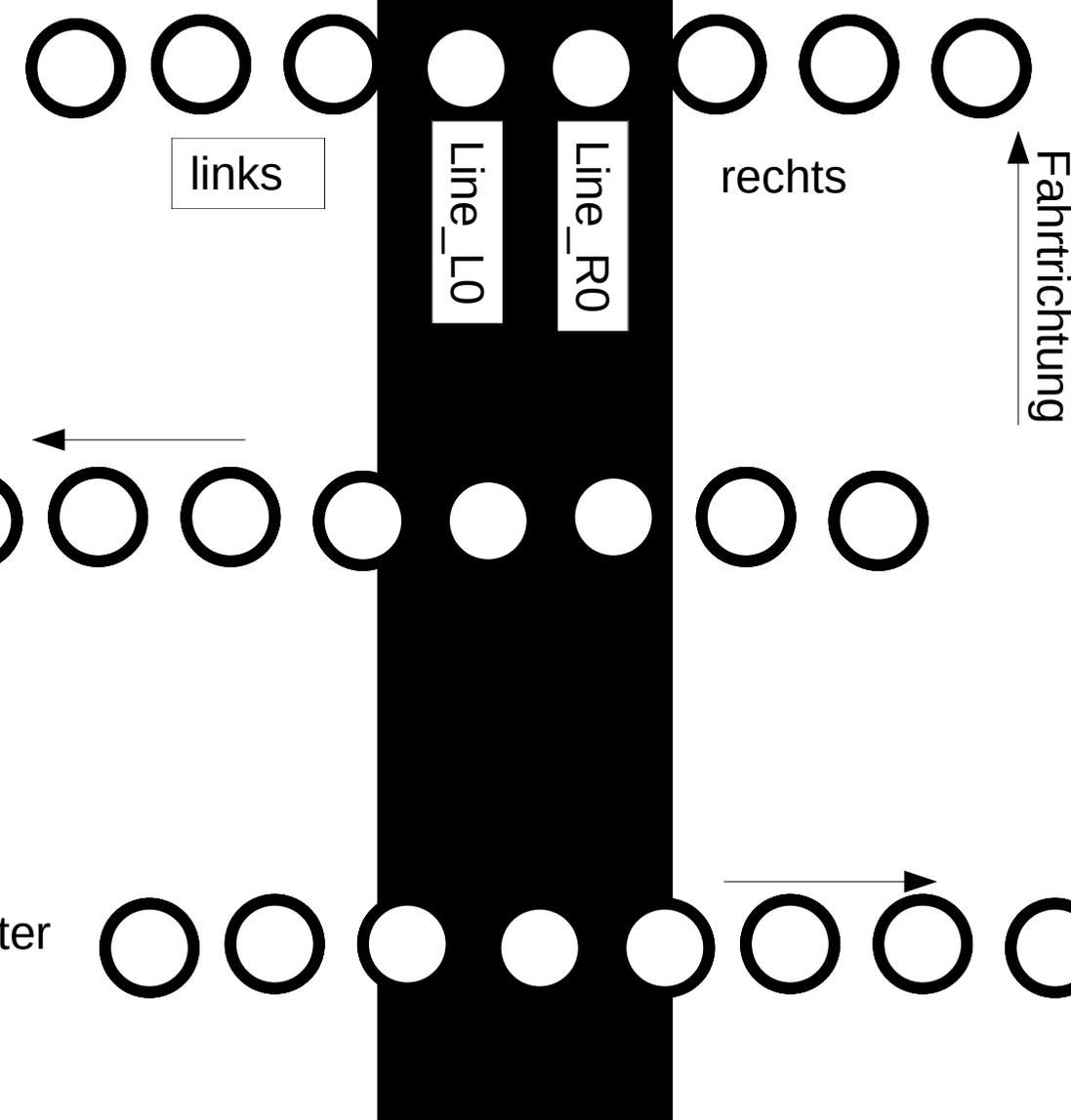
Bit 3 hat den Stellenwert 8
Bit 4 hat den Stellenwert 16

Stellenwerte von Dezimalzahlen



1 Achter
0 Vier
1 Zweier
1 Einer

3 Hunderter
0 Zehner
5 Einer



```

Test_Linesensor.cpp* Interrupt_Service_Routines.h USART_Functions.h Init.h
Test_Linesensor.cpp
49 #define Line_L3 Analogue_value[0] //Line sensor left outside
50 #define Line_L2 Analogue_value[1]
51 #define Line_L1 Analogue_value[2]
52 #define Line_L0 Analogue_value[3] //Line sensor left middle
53 #define Line_R0 Analogue_value[4] //Line sensor right middle
54 #define Line_R1 Analogue_value[5]
55 #define Line_R2 Analogue_value[6]
56 #define Line_R3 Analogue_value[7] //Line sensor right outside

100%
94 while (1) {
95     Data_Visualizer();
96     _delay_ms(5);
97     if (Line_L0 < Threshold) Line_L0_digit = 0b00010000; //The forth line sensor, from the left side, sees white
98     else Line_L0_digit = 0; //It is black
99     if (Line_R0 < Threshold) Line_R0_digit = 0b00010000; //The fifth line sensor, from the left side, sees white
100    else Line_R0_digit = 0; //It is black
101 }
102 }
103
104 /*****
105 //Other functions
106
107 //Data beeing transmitted to the Data Visualizer
108 void Data_Visualizer (void) {
109     Data[0] = Line_L3; //Line sensor left outside
110     Data[1] = Line_L2;
111     Data[2] = Line_L1;
112     Data[3] = Line_L0; //Line sensor left middle
113     Data[4] = Line_R0; //Line sensor right middle
114     Data[5] = Line_R1;
115     Data[6] = Line_R2;
116     Data[7] = Line_R3; //Line sensor right outside
117     Data[8] = Line_L0_digit;
118     Data[9] = Line_R0_digit;
119 }
100%

```

Data Visualizer

Configuration

DGI Control Panel

Serial Port Control Panel

USB-SERIAL CH340 (COM) Disconnect

DTR RTS

Open Terminal

Autodetect protocols

Baud rate: 78600 Parity: None Stop bits: 1 bit

Terminal 10

89	87	250	249	77	68	97	63	0	8
91	91	251	249	78	69	98	64	0	8
93	92	251	249	79	70	98	65	0	8
93	91	250	249	79	69	97	64	0	8
91	90	250	249	77	68	97	63	0	8
88	87	250	248	75	66	96	63	0	8
87	86	250	248	75	66	96	63	0	8
88	87	250	248	77	67	97	63	0	8
91	90	250	249	78	69	98	64	0	8
92	91	251	249	79	69	98	65	0	8
93	91	250	249	79	69	98	64	0	8
91	90	250	249	77	68	97	63	0	8
89	88	250	249	76	67	96	63	0	8
87	86	250	248	75	66	96	63	0	8
88	87	250	249	76	67	97	63	0	8
90	89	250	249	78	68	97	64	0	8
91	91								

Clear Add \r\n Hexadecimal Values Show Timestamp

Automatically Scroll to End

Montage und Programmierung
eines Roboters für
den Hessen SolarCup
Disziplin: SolaRobot
Teil 2.5: Ultraschall

Von Charlotte und Andreas