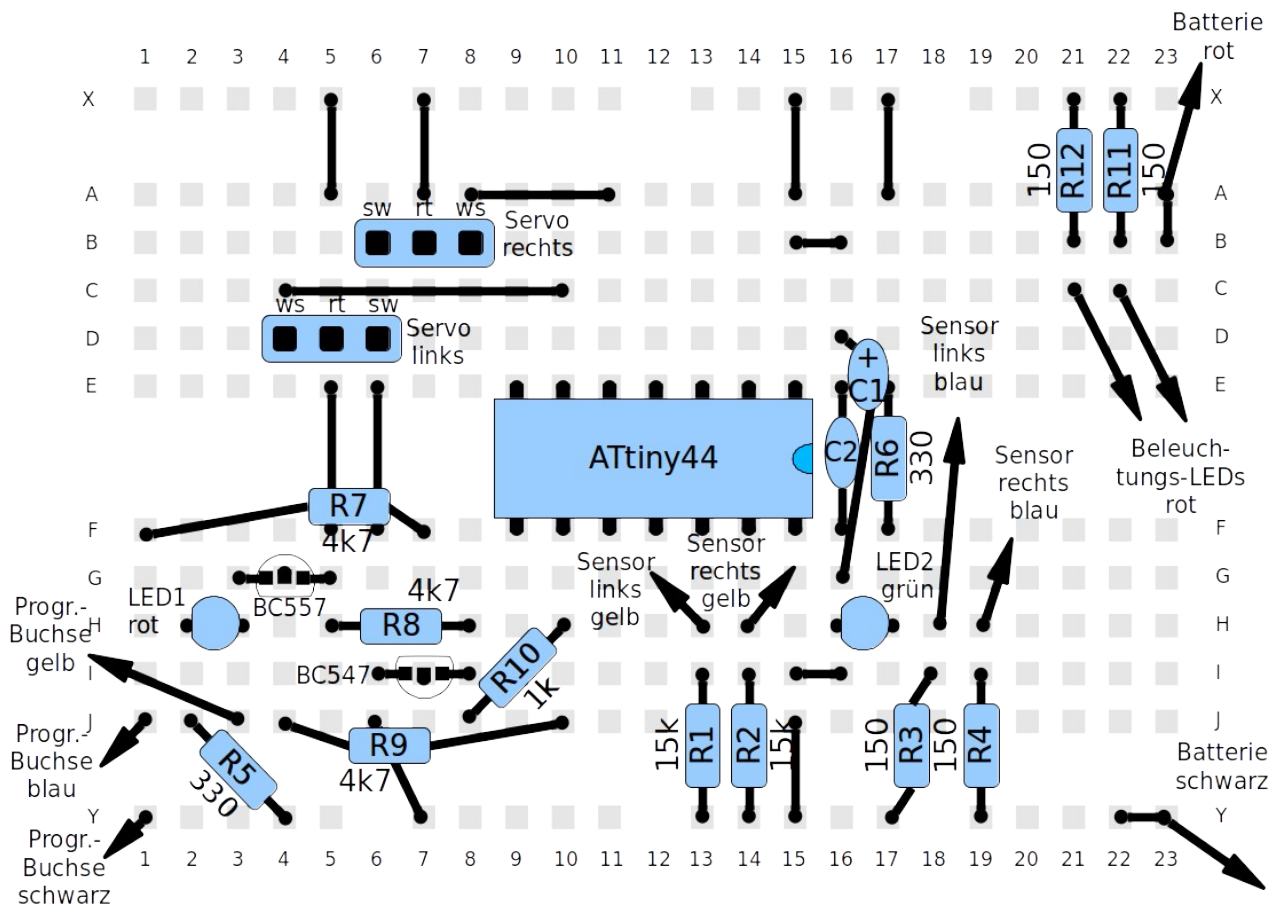


Bestückungsplan der Robotersteuerung



Die Pfeile stellen Kabel dar, die unter anderem zu den Sensoren führen. Sie werden jeweils an den Startpunkten der Pfeile in das Experimentierbrett gesteckt.

Es sind nicht alle Kabel dargestellt. Wenn nicht anders angegeben, kommen rote Kabel irgendwo in die X-Zeile und schwarze Kabel irgendwo in die Y-Zeile.

Achtung: dies gilt nicht für die roten Kabel der Beleuchtungs-LEDs! Sie kommen in C21 und C22!

Die Kabel der Motoren (Servos) werden auf die dreipoligen Steckerleisten gesteckt, hier im Plan mit „Servo rechts“ und „Servo links“ bezeichnet.

Achtung: sie müssen unbedingt richtig herum eingesteckt werden, also das schwarze Kabel bei der Markierung „sw“ und das weiße (bei manchen Servos ist es orange) bei „ws“.

Kondensator „C1“ **muss ebenfalls richtig herum** eingesteckt werden. Ist es ein ockerfarbener Typ (Tantal), bezeichnet die Markierung den Pluspol und muss daher nach oben in D16; bei allen anderen Typen bezeichnet die Markierung den Minuspol und muss nach unten in G16. (Die Farben gelten nicht grundsätzlich überall, sondern nur bei den von uns verwendeten Typen).

Fragen oder Probleme? 🤖

=> E-Mail an robo-ag@roboretto.de 😊

Programm zum einfachen Linienfolger

```
void folgeLinie( void )
{
    uint helligkeit;

    while ( 1 )
    {
        helligkeit = SensorLinks();

        if ( helligkeit > 50 )
        {
            setzeMotorLinks( 1 );
            setzeMotorRechts( 0 );
        }
        else
        {
            setzeMotorLinks( 0 );
            setzeMotorRechts( 1 );
        }
    }
}

int main( void )
{
    initHardware();

    folgeLinie();

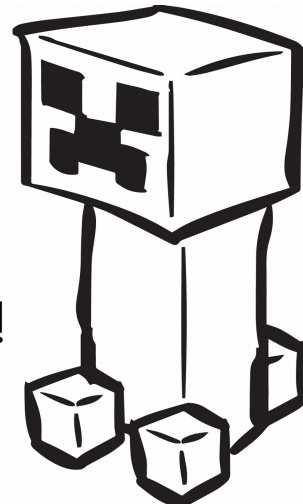
    // Endlosschleife, damit die LED und die Motoren weiter arbeiten
    while (1)
        ;

    return 0;
}
```

Anzeige ☺

Pixelcreeper.de

Für den Spielspaß!



Programm zum Linienfolger mit 2 Sensoren

```
void folgeLinieProVersion( void )
{
    // Variablen müssen ganz am Anfang einer Funktion erstellt werden

    uint korrektur = 0;    // hier müssen wir einfach irgendeinen Wert setzen,
                          // damit der Compiler nicht meckert. Wir nehmen 0,
                          // so dass erst mal nicht korrigiert wird.

    uint hellLinks;
    uint hellRechts;

    // Endlosschleife
    while ( 1 )
    {
        // beide Sensoren einlesen
        hellLinks = SensorLinks();
        hellRechts = SensorRechts();

        // sind wir mit beiden Sensoren auf der Linie, also der
        // linke UND (&&) der rechte Sensor dunkel?
        if ( ( hellLinks <= 50 ) && ( hellRechts <= 50 ) )
        {
            // ja, also geradeaus
            setzeMotorLinks( 1 );
            setzeMotorRechts( 1 );
        }
        else
        {
            // sind wir links, halb auf der Linie?
            if ( ( hellLinks > 50 ) && ( hellRechts <= 50 ) )
            {
                // ja, also nach rechts korrigieren.
                setzeMotorLinks( 1 );
                setzeMotorRechts( 0 );

                // merken, dass wir zu weit links waren
                korrektur = 1;
            }
            else
            {
                // sind wir rechts, halb auf der Linie?
                if ( ( hellLinks <= 50 ) && ( hellRechts > 50 ) )
                {
                    // ja, also nach links korrigieren
                    setzeMotorLinks( 0 );
                    setzeMotorRechts( 1 );

                    // merken, dass wir zu weit rechts waren
                    korrektur = 2;
                }
                else
                {
                    // hier kann nur noch hell / hell vorliegen, also dass
                    // wir komplett weg von der Linie sind.
                    // Jetzt kommt es darauf an, auf welcher Seite wir die Linie
                    // verlassen haben. Zum Glück haben wir uns das in der
                    // Variable "korrektur" gemerkt!
                }
            }
        }
    }
}
```

