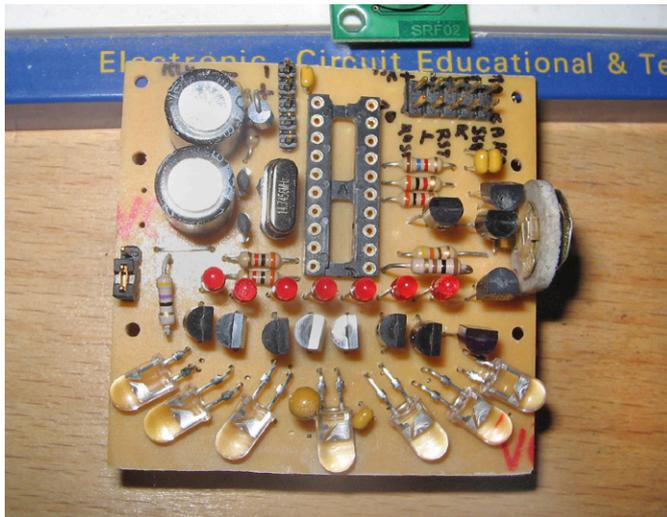


Schneller IR-Roboter-Bumper

Erkennt Hindernisse in allen Richtungen



Von **Ralf Schmiedel**

Die schnelle berührungslose Kollisionserkennung für ebenso schnelle Roboter zeigt nicht nur an, dass ein Hindernis auftaucht, sondern auch, in welcher Richtung sich das Hindernis befindet. Bei dem Projekt spielen Schaltung und Software trickreich zusammen.

Den IR-Bumper kann man sich wie einen umgekehrten Scanner vorstellen. Eine halb-kreisförmige IR-LED-Zeile sendet einen Scan aus, die IR-Dioden werden dabei von einem Mikrocontroller nacheinander angesteuert. Ein IR-Sensor mit großem Blickwinkel fängt das IR-Licht auf, das von einem Hindernis reflektiert wird. Der IR-Sensor hat nur einen digitalen Ausgang (Licht / kein Licht); ein Mikrocontroller muss anhand der zeitlichen Information ermitteln, zu welcher IR-Diode das Licht gehört. Man könnte sich nun selbst eine

passende Synchronisation überlegen, doch warum nicht ein Protokoll benutzen, das es schon gibt? Wenn man die IR-LEDs mit UART-Timing blitzen lässt und den Ausgang des IR-Sensors mit dem UART-Eingang (RXD) des Controllers verbindet, dann empfängt man Bits, die jeweils einer IR-Diode zugeordnet sind. So kann man sogar die Richtung von mehreren Hindernissen gleichzeitig erkennen. In diesem Projekt sind sieben Dioden verbaut, die je einem Bit im UART-Protokoll 8N1 entsprechen (**Bild 1**).

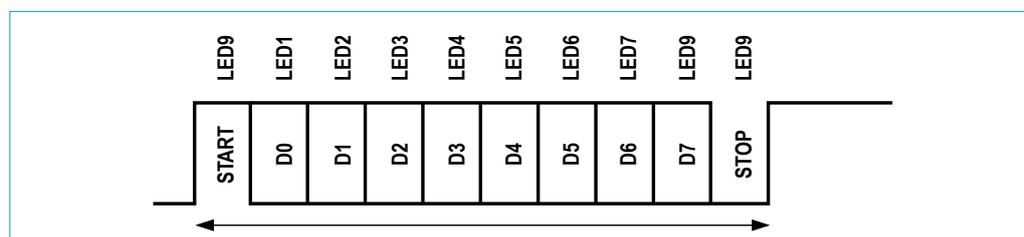


Bild 1.
Zuordnung der Dioden zum
8N1-Datenprotokoll.

Es müssen noch drei weitere Bits erzeugt werden (Startbit, 8. Datenbit und Stoppbit). Dies übernimmt eine weitere IR-LED, die direkt auf den IR-Sensor gerichtet ist. Sie hat darüber hinaus eine spezielle Funktion, nämlich den Verstärker der IR-Sensoren einzustellen. Dazu mehr im nächsten Abschnitt.

Der Bumper wird über die gleiche UART-Schnittstelle mit dem Host-Controller des Roboters verbunden und gibt darüber die Ergebnisse der Hindernismessung aus. Der Host-Controller wiederum spricht den Bumper-Controller mit einfachen ASCII-Befehlen an, um seine grundlegenden Parameter einzustellen.

IR-Dioden

Der IR-Bumper kann wahlweise für einen Scan-Bereich von 90° oder 180° bestückt werden. Nach dem gleichen Prinzip lassen sich aber auch Einzelsensoren mit einem Controller im SO-8-Gehäuse oder auch 360°-Sensoren mit entsprechend größerem AVR-Controller ausführen.

Als IR-Sensor kommt der sehr schnelle TSOP7000 von Vishay [1] zum Einsatz. Es handelt sich um einen Fernsteuerempfänger, der nicht wie die meisten anderen TSOP-Modelle mit 30...56 kHz arbeitet, sondern mit einer Modulationsfrequenz von 455 kHz. Wie alle Fernsteuerempfänger besitzt auch der TSOP7000 einen AGC-Verstärker (automated gain control) in der Eingangsstufe hinter dem Fotoverstärker. Dieser passt die Verstärkung an das sehr unterschiedlich starke Signal einer Fernbedienung an und verhindert das Über-

steuern der nachfolgenden Demodulationsstufe. Für unsere elektronische „Stoßstange“ ist die AGC-Funktion allerdings kontraproduktiv. Die Empfindlichkeit würde je nach Diode bis zu 20 m Reichweite aufgedreht, der Bumper so immer ein Hindernis anzeigen. Andererseits bietet die Eingangsstufe eine extrem schnelle und schmalbandige Empfangstechnik bei hoher Empfindlichkeit. Das sind Eigenschaften, die wir hier benötigen.

Um die AGC-Funktion unwirksam zu machen und als einstellbare Verstärkerstufe mit konstanter Verstärkung zu nutzen, wird die AGC mittels eines „Blitzes“ wohldosiert heruntergefahren. Der Blitz wird von ein bis zwei zusätzlichen IR-Sendedioden erzeugt, die direkt auf den Sensor gerichtet sind. Die Verstärkung wird durch ein PWM-Signal (und ein Trimpoti) justiert. Da die Reaktionszeit der AGC recht langsam ist (mehrere 100 ms), ergibt sich bei regelmäßigem Blitzen eine konstante Verstärkung.

Als Sendedioden werden schnelle IR-Dioden von Vishay [2][3] genutzt, je nach gewünschtem Abtastwinkel mit einem Empfangsbereich von 10° (90° Abtastwinkel) oder 22° (180° Abtastwinkel). Damit ergibt sich eine durchgehende Abtastung (**Bild 2**). Der TSOP7000 besitzt einen Empfangswinkel von 90°, sodass für eine 180°-Abtastung je zwei dieser Sensoren in Open-collector-Schaltung (geodert) bestückt werden müssen.

Als Controller wird der AVRtiny2313 verwendet, der über eine UART-Schnittstelle verfügt und sehr preisgünstig ist.

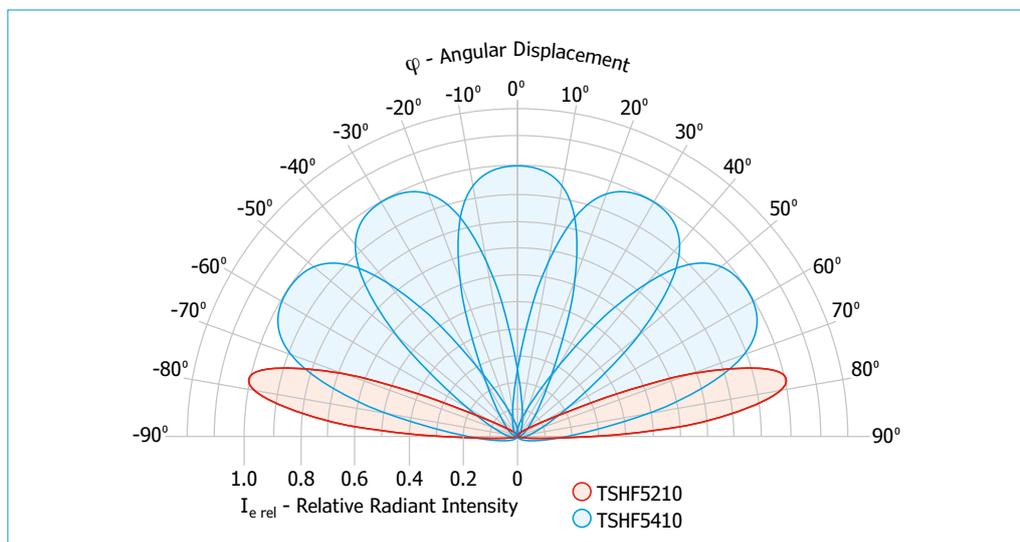


Bild 2. Abstrahlcharakteristik der IR-Dioden für 180°.

Die Schaltung

In **Bild 3** haben wir nun endlich die Schaltung vor uns. An den Controllerausgängen PB0...PB1 und PD2...PD6 sind die Sendedioden LED1...LED7 angeschlossen.

IR-Sende-Dioden gemultiplext, sodass nur ein zusätzlicher Port benötigt wird. Die „normalen“ roten LEDs an den Basen der Transistoren werden über PB6 auf Masse gelegt und leuchten, wenn die entsprechende Pufferstufe

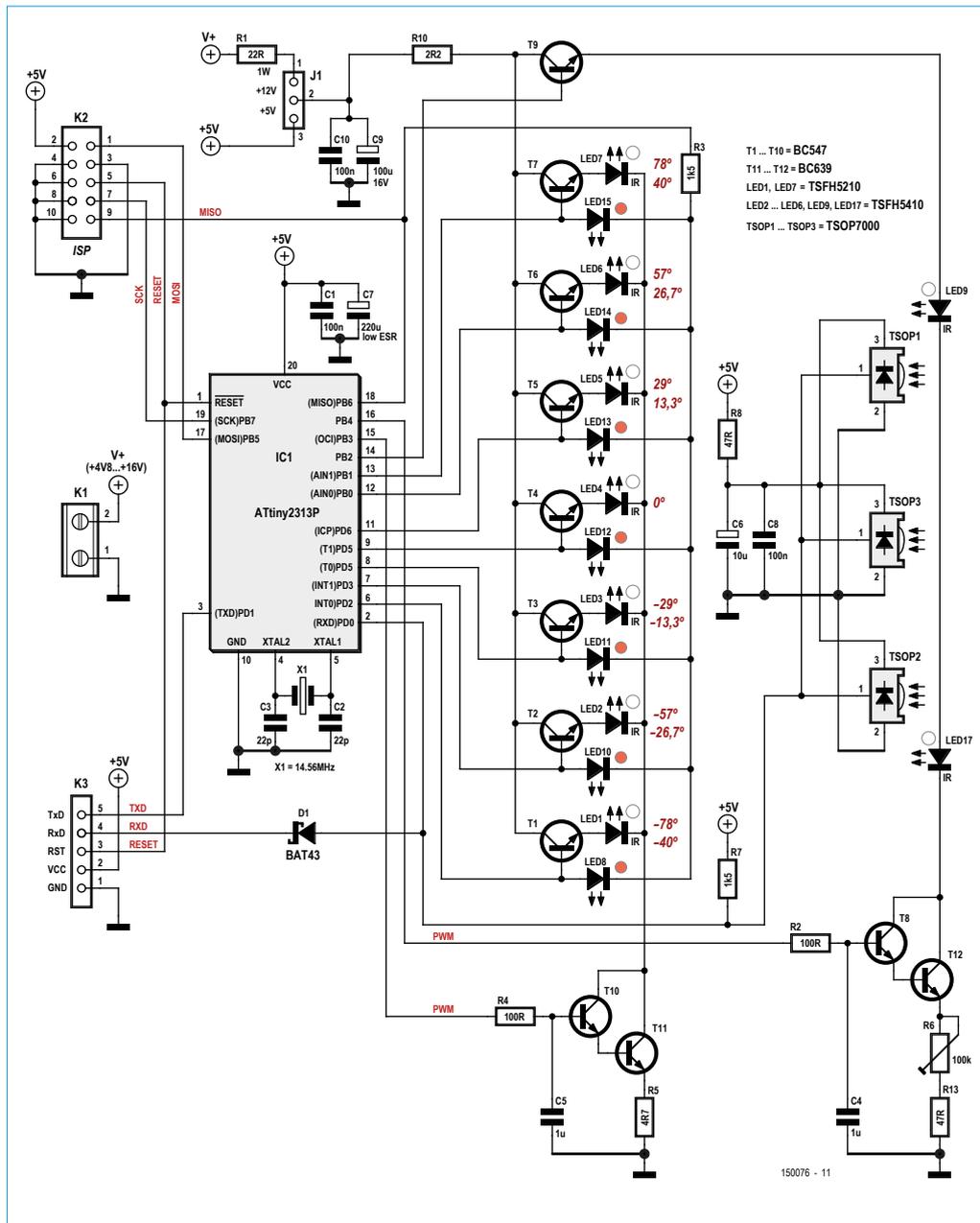


Bild 3. Die Schaltung des Bumpers.

Der Strombedarf dieser Dioden übertrifft aber die Möglichkeiten des Controllers, so dass pro IR-LED(-Pärchen) ein Kleinsignaltransistor erforderlich ist.

Um auch ohne angeschlossenen PC etwas zu sehen, gibt es sieben rote LEDs zur Anzeige der Scanergebnisse. Diese sind mit den

angesteuert wird. Man kann die roten Anzeigedioden auch gänzlich abschalten, indem man PB6 nicht auf Masse legt (und so Energie sparen).

Portpin PB3 gibt ein PWM-Signal aus, das den Strom durch die Sendedioden bestimmt. Das Signal wird durch den Tiefpass R4/C5 geglä-

Technische Daten

- Modulationsfrequenz der Sendedioden: 455...460 kHz
- Sendefrequenz eines Scans: 19,2 kHz oder 38,4 kHz
- Scandauer: 2 ms
- Scanfrequenz (freilaufend): ca. 105 Hz
- Reichweite: 20 cm bis über 2 m, abhängig vom Reflexionsverhalten des Hindernisses. Nur bei sehr dunklen Objekten deutliche Minderungen. Verdopplung der Reichweite durch Bündelung der 10°-Dioden bei der 90°-Version möglich.

tet und steuert so als analoge Spannung die Darlingtonstufe T10/T11 (BC547/BC639), die mit Fußwiderstand R5 als Stromsenke für die IR-Sendedioden fungiert. Der Strom durch die IR-LEDs berechnet sich nach

$$I = (V_{CC} \times \text{Tastverhältnis} - 2 \times U_{be}) / R5$$

An den gemeinsamen Kathoden der roten LEDs ist mit R3 ein Strombegrenzungswiderstand geschaltet. Übrigens: Basiswiderstände an den Puffertransistoren sind wegen der Stromeinstellung der Darlingtonstufe nicht erforderlich, der Basisstrom stellt sich automatisch ein und trägt auch ein klein wenig zum Senden bei. So spart man nicht nur Bauteile, sondern auch noch etwas an Energie! Die beiden Blitzdioden LED9 und LED17 erhalten ihren Strom durch eine ähnliche Anordnung wie die anderen Scandioden, nämlich via PB4 und die Darlington-Stromsenke um T8/T12. Der Blitzstrom und damit die Empfindlichkeit der TSOPs lässt sich am Trimpoti R6 einstellen.

Wie schon beschrieben, sind die IR-Empfänger über ihre Open-collector-Anschlüsse „geodert“. Ihr gemeinsames Signal gelangt (mit R7 als Pull-up-Widerstand) zu PD0 des Controllers. Die UART-Schnittstelle des Controllers „sammelt“ an RXD aber nicht nur die Daten der IR-Sensoren, sondern dient auch der Kommunikation mit dem Host-Controller (K3). D1 isoliert dabei die RXD-Leitung vom Host von den TSOP-Signalen. Auch die Betriebsspannung des Controllers muss vom Host geliefert werden. Da wir es mit recht hohen Sendeströmen bis 0,5 A zu tun haben, sorgt ein Elko mit niedriger ESR (C7) für eine stabile Spannung.

Software

Damit sind wir schon beim Kapitel Software! LED9/LED17 sorgen mit einem Blitz nicht nur für die Verstärkungseinstellung der TSOPs, sondern erzeugen damit auch das Startbit für das 8N1-Protokoll. Die ersten sieben Datenbits enthalten die Scan-Informationen, das achte Bit (ist immer „1“) und das Stoppbit wird wiederum von einem Blitz erzeugt.

Der Host kommuniziert mit dem Bumper-Controller im alten ASCII-Code, der nur 127 Zeichen umfasst, also nur sieben Bit in Anspruch nimmt. Das achte Bit des Protokolls ist in dieser Kommunikation immer „0“. So kann die Software die Signale vom IR-Sensor und vom Host unterscheiden.

Einen kleinen Schönheitsfehler gibt es allerdings. IR- und Host-Empfang sind zeitlich nicht aufeinander abgestimmt, die Zeichen könnten sich also überlagern und stören. Das Problem lässt sich aber lösen. Der Host muss zunächst einen Prompt > (ASCII 62) erhalten, die Erlaubnis, einen Befehl senden zu dürfen. Der Host fordert den Prompt durch mehrfaches Senden des ESC-Zeichens (ASCII 27) an. Nach etwa fünf ESC-Zeichen sollte es ziemlich sicher sein, dass dem Host ein Prompt vorliegt (wenn nicht, wiederholen). Sobald der Prompt beim Host eintrifft, gehört die Schnittstelle dann exklusiv dem Host und es wird nicht mehr gescannt. Wenn der Scan dann wieder läuft, muss der Host in der Regel nicht mehr schreibend auf den Bumper zugreifen und braucht nur noch die Werte zu lesen, die gesendet werden.

Das Programm wurde in BASCOM erstellt. Nur für das Senden über die IR-Dioden wurde sehr einfacher Assemblercode eingesetzt, um 455 kHz zu erreichen. Die Trägerfrequenz von 455 kHz erlaubt laut Datenblatt eine Modulationsfrequenz von 20 kHz, was dann 19,2 kBit/s als üblicher Datenrate entspräche. Ich habe aber festgestellt, dass auch 38,4 kBit/s locker verarbeitet werden, und dies in der Software implementiert. Auch andere Frequenzen sind möglich (siehe Programmkommentar).

In einer Schleife wird die Rx-Leitung der UART-Schnittstelle abgefragt und entsprechend darauf reagiert. Es werden die in **Tabelle 1** aufgeführten Kommandos/Funktionen unterschieden. Im Assemblerteil wird ganz einfach mit NOP-Anweisungen je nach Quarz das Timing eingestellt (siehe Quelltext [2]). Für eine andere Quarzfrequenz muss

Tabelle 1. Befehlsübersicht.		
Kommando	Funktion	Sinnvolle Werte
A	Stellt PWM_A des Controllers ein. Diese steuert die Sendeleistung.	30..130
B	Stellt PWM_B des Controllers ein. Diese steuert die Verstärkung des „ehemaligen“ AGC.	30..130
M	Set Mode: Einzelne Bits steuern folgende Funktionen, Rest unbenutzt. Bit2: PWM wird doppelt inkrementiert, daher doppelte Scan-Geschwindigkeit bei Progressive Scan. Bit4: 1= rote LEDs an. Bit4: 0= rote LEDs aus.	
R	Radar, frei laufender Progressive Scan Mode.	Ausstieg mit ESC (ASCII 27), ggf. mehrfach erforderlich.
F	Free running. Freilaufende Scans.	Ausstieg mit ESC (ASCII 27), ggf. mehrfach erforderlich.
W	Stellt die Wartezeit zwischen 2 Scans in ms ein (mindestens 2 ms).	0..255
P	Progressive Scan, einmaliger Scan mit steigender Reichweite.	
S	Scan once, scannt einmal und antwortet dann mit dem Ergebnis.	
I	Set Initial command: Eines der obigen Kommandos wird nach dem Start automatisch ausgeführt, alle anderen ASCII-Werte führen zu einem Prompt „>“ für eine Eingabe	

Die Kommandos dürfen auch Kleinbuchstaben sein.

man nur den Kommentaren im Programm folgen.

Um einen Scan zu senden, werden zunächst die AGCs der TSOP7000 „geblitzt“, um sie auf die gewünschte Empfindlichkeit einzustellen. Dann senden die Blitz-Dioden zunächst ein Startbit, danach senden die sieben Scan-Dioden jeweils ein Bit des UART-Bytes, gefolgt von einem achten Bit und einem Stoppbit über die Blitz-Dioden. Der Empfang eines Scans ist denkbar einfach: Der TSOP7000 ist am Rx-Eingang der UART-TTL-Schnittstelle angeschlossen, sodass BASCOM direkt auf ein Byte zugreifen kann.

Aufbau und Einstellung

Noch ein paar Tipps zu den IR-Dioden und -Empfängern. Wegen der hohen Sendefrequenz sind die hier gezeigten sehr schnellen (oder vergleichbare) IR-Dioden erforderlich, um die mögliche Reichweite auszuschöpfen.

Die sehr verbreitete IR-Diode LD274 mit 20°-Öffnungswinkel ist für diesen Zweck zu langsam und funktioniert nur bei kurzen Reichweiten bis circa 30 cm. Die Reaktionszeit/Flankensteilheit reicht einfach nicht, um mehr Sendeleistung zu realisieren.

Die Abtastcharakteristik des TSOP7000 findet man im Datenblatt, es ist eine Keule mit etwa 90°-Öffnungswinkel. Die 90°-Bestückungsvariante hat daher ein Abtastfeld von etwa dieser Form. Für die 180°-Version kommen zwei TSOP7000 zum Einsatz, die nach 45° und -45° ausgerichtet sind. Die Randschwäche der Empfänger hat zur Folge, dass der Sensor dann nicht nur am Rand, sondern auch in der Mitte etwas schwächer ist.

Im Schaltplan sind für einen Scanwinkel von 180° als äußerste Dioden die TSHF5210 (10°) eingesetzt worden, die nur den halben Winkel der TSHF5410 (22°), dafür aber eine höhere Leuchtdichte aufweisen. Dadurch werden die

am Rand schwächeren Empfangseigenschaften der TSOP ausgeglichen und es ergibt sich ein grob betrachtet rechteckiges Scanfeld vor dem Sensor. Man kann aber auch am Rand die TSHF5410 einsetzen.

Mit den zwei TSHF5210 an den äußeren Positionen kann man zwar die Randschwäche etwas ausgleichen und erreicht ein fast rechteckiges Abtastfeld, das innen leicht eingewölbt ist. Möchte man auch etwas für die Mitte tun und das 180°-Feld mehr halbkreisförmig abtasten, kann man einen dritten TSOP hinzufügen, der wie bei der 90°-Version nach vorne schaut. Aus der Überlagerung ergibt sich dann ein mittig weiter reichendes, fast halbkreisförmiges Messfeld.

Bild 4 und [5] zeigen Aktivierungskurven mit den Hindernissen (von innen nach außen) Kugelschreiber, Hand, weißes Blatt Papier bei unterschiedlichen PWM-Einstellungen. Das Messfeld ist nicht ganz symmetrisch, weil

Beim Aufbau der Platine ist darauf zu achten, dass TSOP7000, LED9 und LED17 (nur bei 180°-Scanwinkel) von unten verlötet werden. Die Platine muss optisch undurchsichtig sein. Hier kann ein Cu-Layer, Metallklebeband und ganz simpel ein schwarzer Edding helfen. Die IR-Dioden muss man mit einer Zange sorgfältig rechtwinklig zum Gehäuse um 90° abwinkeln, nur so werden die Winkel der Dioden richtig eingehalten.

Hat man alle Bauteile glücklich auf (und unter) der Platine angebracht, kann man die TTL-Schnittstelle mit dem Hostcontroller verbinden und die Betriebsspannung Vcc anlegen. Vcc kann dem Hostsystem entnommen und über K3 eingespeist werden. Bei den Sendedioden hat man die Wahl, sie über Vcc (Jumper J1 auf +5 V) oder aus einer unregelmäßigen Spannungsquelle (4,8..16 V, beispielsweise aus einem Akku) zu versorgen. Auf die Sendeleistung hat das keinen Einfluss, da an den

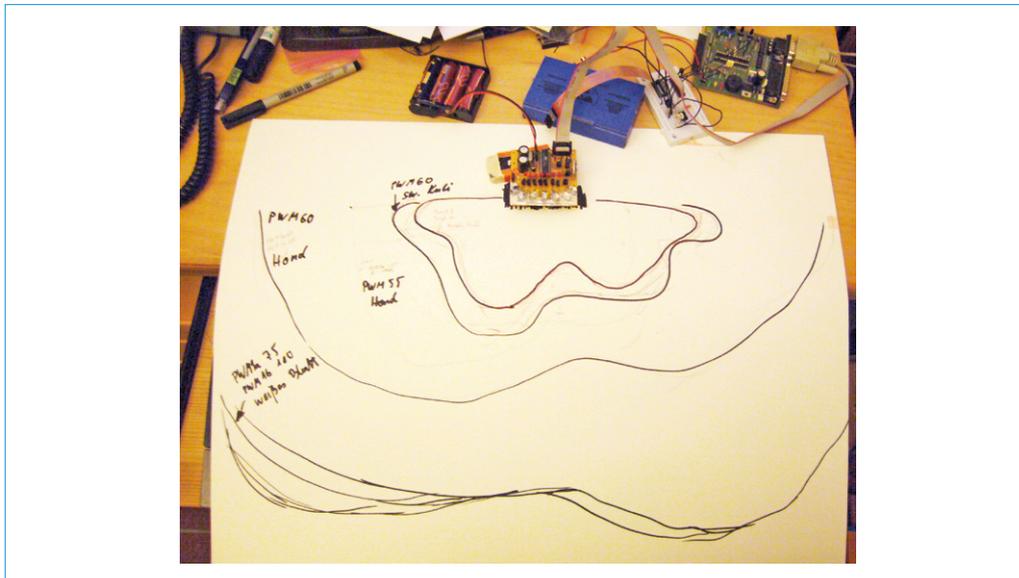


Bild 4. Aktivierungskurven mit unterschiedlichen Hindernissen.

die beiden TSOP7000 offenbar unterschiedlich „geblitzt“ werden, durch Toleranzen aller beteiligten Dioden oder schlicht wegen eines differierenden Abstands der Blitzdioden vom Sensor. Hier sollten die Kurven durch eine mechanische Änderung des Abstands von Diode und TSOP oder elektrisch durch einen Shunt (fest oder Mini-Trimpoti) parallel justiert werden. In der Vorderansicht (**Bild 5**) sieht man den unterschiedlichen Abstand der Blitzdioden zu den TSOP nach der mechanischen Justierung des Messfeldes.

Emittieren von T1 bis T7 immer nur rund 4,3 V anliegen ($V_{cc} - U_{be}$). Die optionalen R1 und R10 reduzieren die Verlustleistung über T1 bis T7 bei sehr schnellen Dauer-Scans und maximaler Reichweite. Sie können in der Regel durch Brücken ersetzt werden.

Für erste Versuche empfiehlt es sich, einen PC mit einem Terminalprogramm (zum Beispiel Hyperterminal) mit den Einstellungen 8N1 zu verwenden, die TTL-UART-Schnittstelle verbindet man über einen USB/TTL-Konverter oder

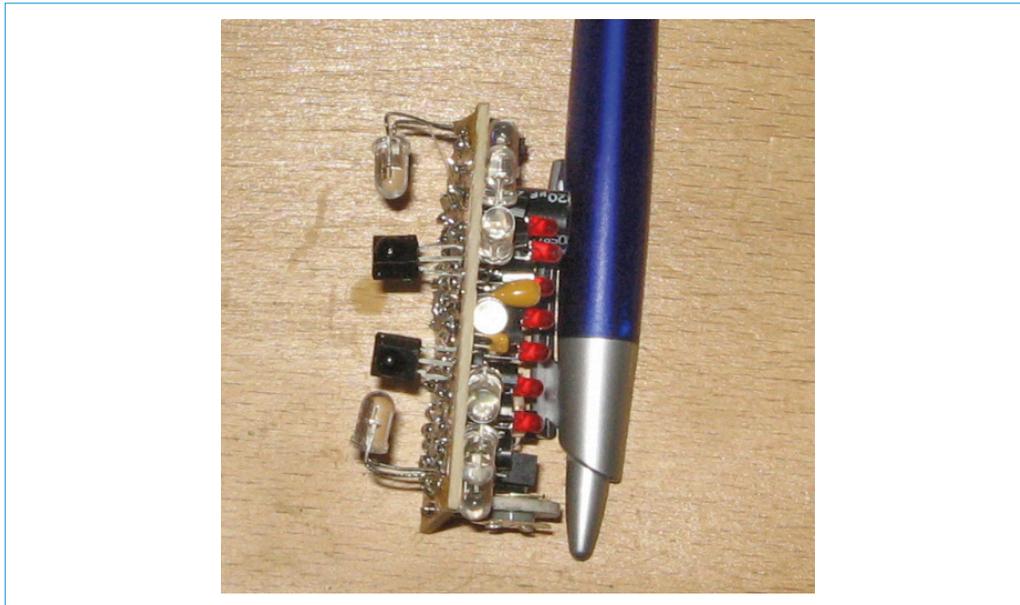


Bild 5.
Links sind die Blitzdioden mit „kalibrierten“ Abständen zu den TSOP zu sehen.

ein entsprechendes Kabel mit dem PC. Außerdem müssen +5 V an K3, zum Beispiel von der USB-Schnittstelle, bereitgestellt werden. Nach dem Einschalten startet der Controller je nach Wert im „Initial Command-Register“ entweder mit einer Aktion oder zeigt einen Prompt. Um einen Prompt zu erhalten, muss ein ESC (ASCII 27) gegebenenfalls mehrfach gesendet werden. Jetzt kann man eines der Kommandos aus der Tabelle eingeben.

Bei den Kommandos R und P werden progressive Scans durchgeführt, das heißt, die Empfindlichkeit wird langsam über mehrere Scans hochgefahren.

Als Ergebnis erhält man dann zwei Werte, die Richtung und den PWM-Wert, bei dem es einen Kontakt zu einem Hindernis gab. Da die PWM

nur langsam hochzählt, werden viele Scans benötigt und die Scan-Rate hängt hier von der Anzahl der nötigen Scans ab, bis etwas erkannt wird. Das kann schnell auch in den Bereich 100 ms bis 1 s gehen.

(150076)

Weblinks

- [1] <http://datasheet.octopart.com/TSOP7000.-Vishay-datasheet-595920.pdf>
- [2] www.vishay.com/docs/81313/tshf5210.pdf
- [3] www.vishay.com/docs/81303/tshf5410.pdf
- [4] Software: www.elektormagazine.de/articles
- [5] www.youtube.com/watch?v=AAAdm9u9GZCo