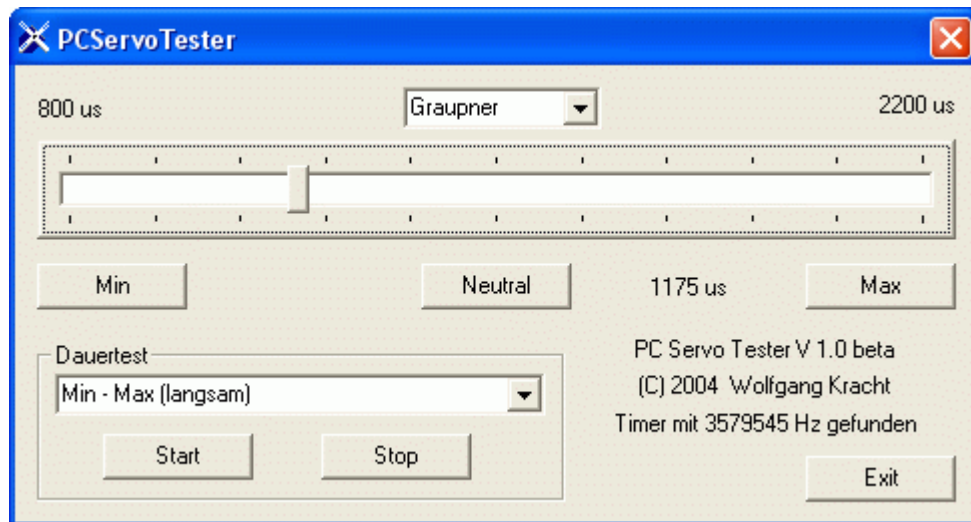


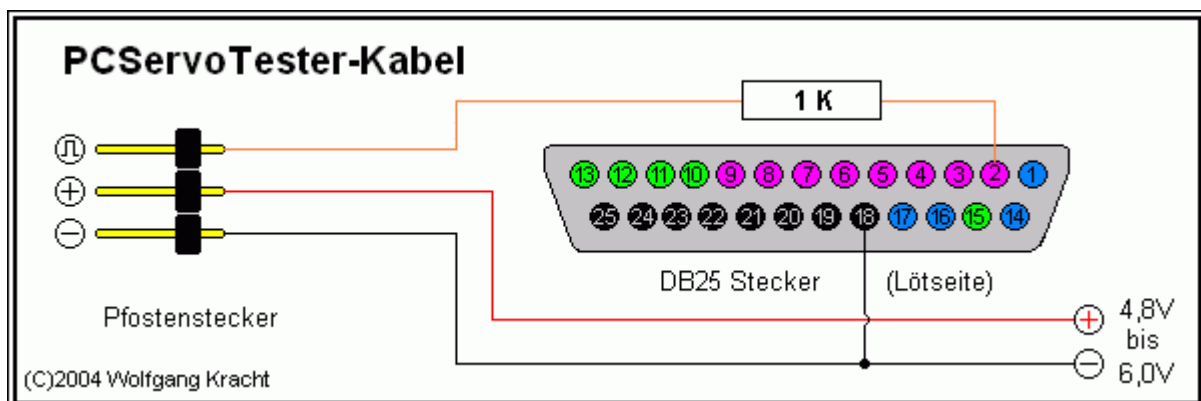
PC als Servotester

Angeregt durch das DOS Programm "MrServo" von [Matthias Rathjen](http://www.logix4u.cjb.net) habe ich einmal versucht etwas Ähnliches als Windows Programm zu entwickeln. Der grosse Nachteil von Bills Fenstern ist, dass das Betriebssystem viel im Verborgenen tut und man somit keine wirklichen Echtzeibedingungen vorfindet, die zur zeitgenauen Ansteuerung des Druckerports von Nöten wären. Unter Windows XP wird dem unbedarften Programmierer durch den Hardware Abstraction Layer (HAL) auch noch der direkte Zugriff auf den Druckerport verwehrt. Dieses Problem lässt sich aber recht einfach mit der "inpout32.dll" von <http://www.logix4u.cjb.net> umschiffen und macht das Programm unabhängig von der verwendeten Windows-Version.

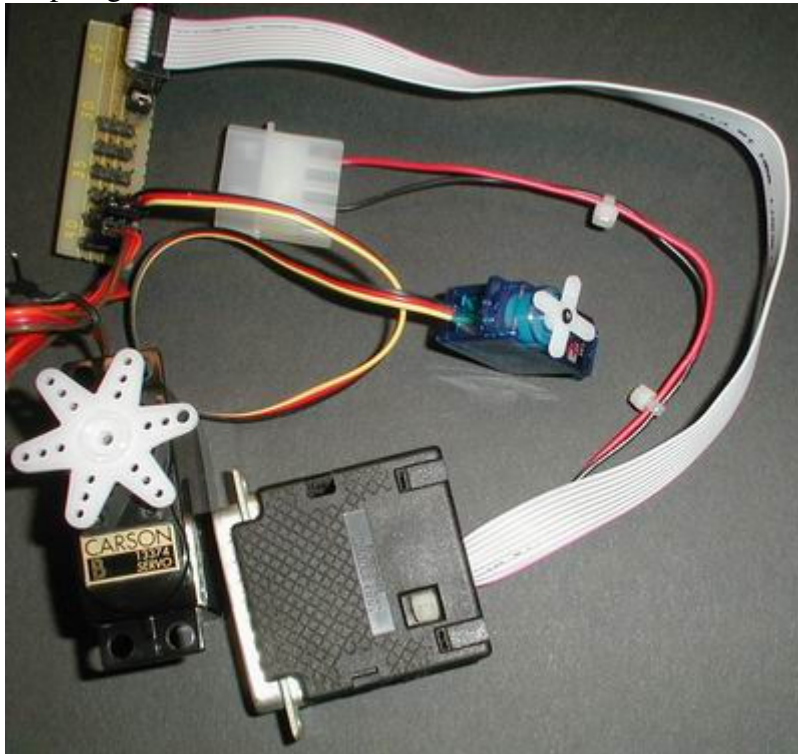
Genug des Vorgeplänckels. Als kleine Machbarkeitsstudie war die erste Version V1.0 beta des Programms recht schnell fertig und sieht wie folgt aus:



Für alle Interessierte, das Programm ist als [PCServoTester.zip](#) online. Die Zip-Datei hat eine Grösse von 79Kb. Diese Version steuert zunächst alle 8 Datenleitungen (Pin 2-9) des Druckerports (LPT1) parallel an. Das Kabel für eine Einkanalversion ist denkbar einfach und günstig ohne aktive Bauelemente aufgebaut. Das folgende Schaltbild ist auch in der Zip-Datei enthalten.



Der Druckerport Anschluss kann nicht die Spannungsversorgung für die Servos liefern. Hierzu bietet sich der PC selbst durch ein herausgeführtes Netzteilkabel oder ein Empfängerakku an.



Da ich gerade Pfostenleisen, Flachbandkabel und DB-25 Stecker mit Gehäuse in Bastelkeller greifbar hatte ist meine Version des Kabels für eventuelle zukünftige Erweiterungen gleich mehrkanalig ausgefallen. Dies wird von dem Programm aber noch nicht weiter unterstützt. Das Stromversorgungskabel wird an einen aus dem PC herausgeführten Netzteilstecker angeschlossen. Es stammt von einem alten defekten CPU-Lüfter und wurde von 12V Anschluss (gelb) auf den 5V Anschluss (rot) umgesteckt und im DB-25 Gehäuse mit den Flachbandkabel

verbunden.

Alle Experimente erfolgen natürlich auf eigenes Risiko. Das Programm ist noch nicht ganz perfekt, erfüllt aber schon gut seinen Zweck. Ab und an gibt es einen Zucker woran man erkennt, dass Windows halt kein Echtzeitbetriebssystem ist. Die Pulszeiten werden recht genau eingehalten. Die Pulswiederholraten von 20 ms sind jedoch nicht immer konstant. Dieses genauer hinzubekommen würde den Programmieraufwand erheblich vergrößern. Getestet habe ich es bislang unter WinXP und Win98SE. Ob die angenommenen Min- und Max-Pulselängen noch den aktuellen Fernsteuerungen entsprechen weiss ich nicht.

Viel Spass beim Ausprobieren ...