Direct Mode Servo - Mehrere Attinys hintereinander abgleichen

Hintergrund

Der auf den MLL Platinen verwendete ATTiny85 Microcontroller wird ohne eigenen Quarz verwendet, da dieser zwei Pins brauchen würde, sowie Platinenfläche und zusätzliche Lötungen und zudem Kosten verursacht. Der verwendete Betriebsmodus hierfür ist „PLL mit chip-internem Oszillator und 16 MHz“. Leider ist die Taktgenauigkeit dieser Lösung nicht sehr hoch und man kann exemplarabhängig Frequenzen zwischen 15,5 und 17,5 MHz messen.

Atmel/Microchip gibt auch Prozeduren zum Trimmen der Taktfrequenz ohne Quarz an, die allerdings entweder einen Atmel Progger mit Kalibriereigenschaften, wie z.B. den Atmel AVRICE, oder einen genauen Frequenzzähler, oder ein Qszilloskop erfordern.

Da das alles für die MLL zu aufwendig ist, wurde ein manuelles (*) Ersatzverfahren entwickelt, dass hier im Weiteren beschrieben werden soll. Hilfreich ist in jedem Fall auch das verlinkte Youtube Video: https://www.youtube.com/watch?v=cKx1rNHql40

(*) „manuell“ heißt, dass DU! ;) arbeiten musst, um es klein und preisgünstig zu bekommen!

Vorbereitungen

Man benötigt an Hardware

- Eine Hauptplatine 101 (Nano) oder 102 (ESP32) - Eine Verteilerplatine z.B. 200 - Zwei oder mehr DM-Servo Platinen 511 (die von Michael/raily74), oder alternativ umgebaute 510 (Anleitung im Wiki) - Eine RGB-LED Kette als finale Heartbeat LED und nachfolgende „Null-Value Detektor LEDs“ Bei Verwendung einer Hauptplatine 102, zusammen mit Heartbeat-Breakout-Boards der ersten Serien (mit Spannungs-Reduktions-Diode) benötigt man eine zusätzliche Heartbeat LED vor der ersten 511 DM-Servo Platine!

Zu installierende Software

Benötigt wird weiterhin eine aktuelle Version des MLL Python Programm Generators von Harold Linke. (Version > 5.3.5e)

Programmieren der ATTiny85

Mit dem aktuellen pyPG wird, unter Zuhilfenahme des 400er Tina Proggers (oder eines anderen geeigneten Programmierwerkzeugs), ein Firmware-Code auf den Tiny geschrieben, der die „Tune

Clock“ Fähigkeit besitzt. (steht im pyPG zur Verfügung!)

Aufbau

Hardware

Man schließt zuerst den Verteiler an die Hauptplatine an. Wenn man eine 102 mit Dioden hat, folgt als erstes eine weitere Heartbeat RGB LED. Dann kommen zwei, oder mehr 511 DM-Servo Platinen mit programmiertem DM-Servo ATTiny85 und auf der finalen 511 gesetztem Terminator. Zuletzt kommt auf den Verteiler ein RGB LED Streifen mit mindestens 10 RGB LEDs.

Software

Im pyPG erzeugt man ein neues Blatt genau für diesen Kalibrierungs-Aufbau. Dieses Blatt sollte folgende Einträge enthalten - Eine Heartbeat LED auf der Hauptplatine - Bei einer 102 mit Diode(n) eine weitere Heartbeat LED auf dem ersten Platz des Verteilers - Pro 511 DM-Servo Platine 3 Servo Beispiel-Animationen die mit DCC Adressen, oder Buttons zu Testzwecken eingeschaltet werden können. Das sind 6 Animationen bei zwei zu trimmenden 511 Platinen, oder 9 Animationen bei drei 511 Platinen usw.! - Eine abschließende Heartbeat RGB LED für den LED Streifen

ATTiny85 vorsortieren

Die Software auf den ATTiny85 kann nicht nur mit vollkommen gleichschnell laufenden Tinys ein gutes Ergebnis erzielen, sondern auch mit dem Prinzip der „Wasserfall Kaskaden“ von „oben nach unten“. Das heißt, ein vorhergehender Tiny darf auch einen Tick schneller laufen, als sein direkter Nachfolger. Nur nicht umgekehrt! (wenn sie genau gleichschnell laufen, ist das aber auch OK)

Diese Anforderung können wir in vielen Fällen schon durch Sortieren der Tinys erreichen! In positiven Fällen muss dann gar nicht nachgetrimmt werden!

Um die optimale Sortierung zu erreichen beginnen wir mit zwei Tinys in den ersten beiden 511 Platinen und terminieren hier die zweite Platine. Wenn jetzt nicht nur der erste, sondern auch der zweite Tiny, im Animations- bzw. Effekt Modus gleichmäßig langsam blinkt und auf dem nachfolgenden LED Streifen die finale Heartbeat LED, auf einer Position weiter hinten, ruhig und gleichmäßig changiert, sowie die LEDs dahinter aus sind, haben wir bereits eine funktionierende Reihenfolge gefunden!

Wenn hingegen die LED auf der zweiten 511 Platine flimmert, oder die finale Heartbeat LED nicht gleichmäßig changiert, oder die LEDs dahinter nicht aus sind und konstant aus bleiben, dann müssen wird die beiden Tinys, in den 511 Platinen, vertauschen!

Eine der beiden Reihenfolgen wird das bessere Ergebnis zeigen!

Nun nehmen wir einen dritten Tiny hinzu. Diesen können wir VOR den beiden vorher sortierten Tinys platzieren, oder DAZWISCHEN, oder DAHINTER.

Auch hier ist eines der Ergebnisse das Beste!

Nachtrimmen der ATTiny85

Wenn wir nun die Tinys sortiert haben, aber immer noch einer der nachfolgenden Tinys manchmal komisch flackert und nicht nur gleichmäßig langsam blinkt, oder die finale Heartbeat LED nicht ausschließlich gleichmäßig changiert, oder eine der LEDs hinter der finalen Heartbeat LED manchmal aufblitzt und nicht kontinuierlich „aus“ ist, dann müssen wir nachtrimmen!

Im pyPG den ersten Tiny trimmen

1. Wir gehen in den pyPG und verbinden uns mit der Hauptplatine.
2. Dann gehen wir auf „Servo Test 2“
3. Dort stellen wir bei „Servo Modul Adresse“ eine 1 ein, wenn wir eine Hauptplatine 101 haben, oder eine 102 ohne Diode(n). Bei einer Hauptplatine 102 mit Diode(n) stellen wir eine 2 ein.
4. Bei „Servo Betriebsart“ wählen wir „Tune Clock“
5. Nun schieben wir den Regler „Servo position“ z.B. auf 20, drücken dann den „Enter“ Button und gucken, was sich verändert.
6. Das Verschieben des Reglers + Enter verändern wir so lange, bis wir das beste Ergebnis erreicht haben.
7. Am WICHTIGSTEN ist hierbei das gleichmäßige Blinken der direkt nachfolgenden 511er Platine!

Im pyPG den zweiten Tiny trimmen

1. Das Trimmen des zweiten Tiny kann nur erfolgen, wenn der erste Tiny so gut getrimmt wurde, dass der zweite Tiny, genau wie der erste Tiny, gleichmäßig langsam blinkt!
2. Um den zweiten Tiny zu trimmen (falls überhaupt erforderlich) wählt man jetzt bei „Servo Modul Adresse“ eine 4 an (oder 5 bei Hauptplatine mit Diode) und wiederholt den Vorgang aus 5.1 so lange, bis der dritte (in diesem Beispiel letzte) 511 Controller gleichmäßig blinkt und auch die finale Heartbeat und die nachfolgenden LEDs keine komischen Effekte mehr zeigen.

Den ersten Tiny evtl. nochmal nachtrimmen

1. Es kann vorkommen, dass der zweite Tiny durch den eigenen Trimm den Kontakt zum ersten Tiny wieder verliert. Man merkt das daran, dass man schlagartig eine Verschlechterung erzeugt und i.d.R. kann man das auch nicht sofort wieder rückgängig machen, weil der zweite Tiny ja nun auch keine Befehle mehr über den ersten Tiny lesbar zugestellt bekommt.
2. Das ist aber leicht zu lösen, indem man die Prozedur aus 5.1 wiederholt und den ersten Tiny wieder besser an den …nun ja selber auch neu getrimmten …zweiten Tiny anpasst!

Nach dem erfolgreichen Trimmen

Wenn man mehrere 511 Controller erfolgreich durchgetrimmt hat sollte man auf jeden Fall bei ALLEN einen „Factory default Reset“ durchführen, da es durch die wirren Bits in der Trimmphase durchaus mal vorkommen konnte, dass ungewollte Programmierungen entstanden sind und gespeichert

wurden!

Mehr als drei Tinys trimmen

Die Prozeduren gelten sinngemäß auch für mehr als drei Tinys und es ist durchaus möglich erfolgreich bis zu sechs 511 Platinen mit ATTiny85 zu trimmen. Die mögliche Anzahl an Sortier- und Trimm-Schritten steigt aber exponentiell an! Einfacher ist es in jedem Fall, nur Cluster bis zu drei 511 Platinen zu trimmen und dann eine WS2811/WS2812 RGB LED dazwischen zu packen, die den Takt aus den 800 kHz Bitrate wieder zurückgewinnen und so für einen weiteren 511er Cluster konsolidieren.