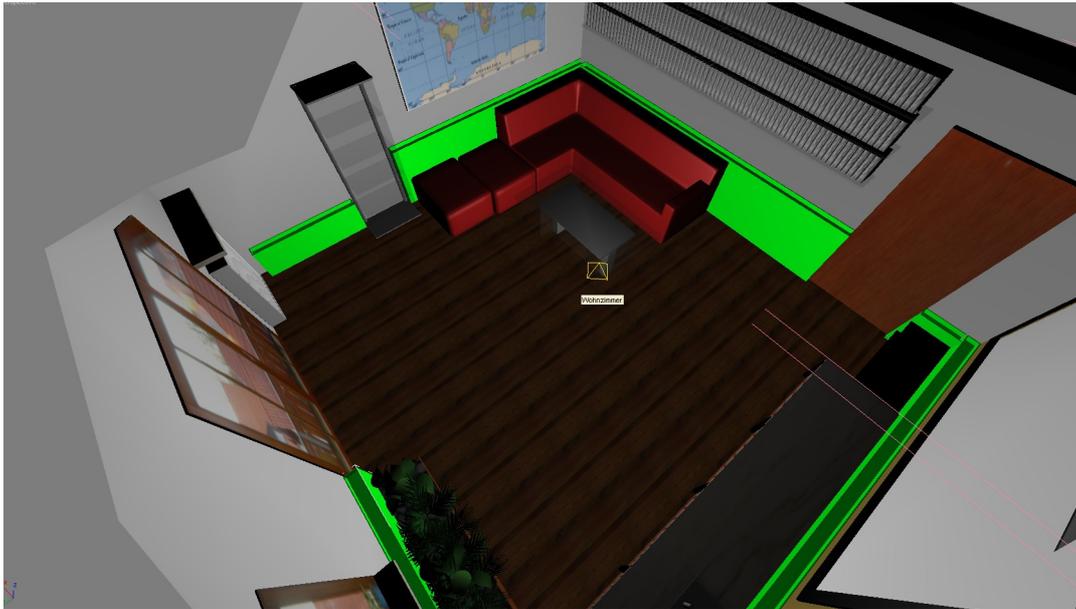


Lastenheft „Wohnzimmer beleuchtung“

Autor : Thomas „JarJar“ Ganshorn

27-01-2009

Einleitung



In diesem Dokument geht es um die Details einer Rechner gesteuerten RGB LED Beleuchtung. Aktuell werden LED Leuchtmittel immer nur als Ersatz für Glühbirnen verwendet und dabei viel Potential verschenkt. Dies will dieses Beispielprojekt ändern.

Die LED's werden zur indirekten Beleuchtung benutzt. Sie sind an allen Wänden des Zimmers angebracht.

Die Beleuchtung besteht aus 28 einzelnen RGB LED Modulen (in diesem Projekt jeweils 50cm lang) die unabhängig voneinander angesteuert werden können.

Die Steuerung der LED Module geschieht über Platinen die jeweils 6 PWM Ausgänge haben und somit bis zu zwei RGB Module steuern können.

Die Platinen sind durch ein Bussystem miteinander verbunden und besitzen eine einstellbare Adresse.

Das Bussystem wird über einen USB Converter mit dem PC verbunden.

Gesteuert werden die Platinen durch entweder eine Masterplatine oder (wie in diesem Fall) einem 5W MiniPC.

Dieser Aufbau ermöglicht ein äusserst flexibles Wohnzimmer mit „KnoffHoff“ vor allem durch die PC Software.

PC Software/Steuerung

Obwohl die PC Software inhouse geschrieben wird, werden ihre Fähigkeiten hier ebenfalls kurz vorgestellt.

Die PC Steuerung ist aufgeteilt in 3 Applikationen. Eine **Steuerapplikation** eine **Editorapplikation** und ein oder mehrere „Fernsteuer“ **Applikationen**.

Steuersoftware

Die **Steuersoftware** bekommt Highlevel Befehle über einen TCP Port. Solche Befehle sind z.b. „starte Lichtsetting Wald, Wechsle für Modul X-Y zur Farbe RGB, Starte Zusätzlichen LichtEffekt Email“

Sie setzt diese Highlevel Befehle um in direkte Steuersignale für die einzelnen Platinen. Das heisst sie sendet jedem Modul zu welche Farbe es für wie lange halten soll bzw. zu welcher Farbe es in welcher Zeit es sich ändern soll.

Editor

Die Editorsoftware dient dazu die verschiedenen Lichtsettings und Effekte zu editieren.

Lichtsettings und Effekte sind grundsätzlich ähnlich. Beide erlauben es für jedes Lichtmodul eine Lichtfarbe bzw. einen Ablauf von Farben zu definieren. Farben können dabei überblendet werden oder auch abrupt geändert werden.

Der Unterschied zwischen Settings und Effekten ist in ihrer Anwendung. Ein Lichtsetting ist die Grundeinstellung, Effekte können auf das Lichtsetting addiert werden.

Als ein Beispiel.

Ein Lichtsetting Wald ist so eingestellt, dass der Raum in ein dunkles grün/blau getaucht ist (die einzelnen Module variieren langsam in ihrer Farbe um eine Art „Blätterrauschen“ im Licht zu simulieren). 3 Module haben eine Art Lagerfeuerflackern und wechseln schnell und eher hektisch ihre Farben in verschiedenen Orange/Rot Tönen.

Ein Effekt Blitz schaltet kurzfristig einzelne LED Module auf helles Weiss.

Während das Lichtsetting selbst stabil bleibt und in der Animation loopt, kann über eine „Fernsteuer“ Applikation ein Blitz abgeschossen werden der den Raum kurzfristig erhellt.

Weitere Beispiele für Settings wären EmailAlert (die einzelnen LED Module ändern ihre Farbe z.b. wellenförmig zu rot und wieder zur Ursprungsfarbe zurück).

Effekte werden genau wie Lichtsettings editiert. Zusätzlich hat hier jedoch noch jedes Modul einen Alphawert, der bestimmt wie stark die Lichtfarbe des Effektes wirkt. Weiterhin kann angegeben werden ob die Lichtfarbe hinzuaddiert oder die des Lichteffektes überschreiben soll.

Fernsteuersoftware

Es kann beliebige Fernsteuerapplikationen geben. Zum Beispiel kann über ein HTML Interface ein Lichtsetting ausgewählt werden. Eine weitere Applikation kann bei Eingang einer Email (je nach Priorität der Mail) bestimmte Effekte starten (wichtige Emails können mit einer roten Welle angezeigt werden, unwichtige mit einer grünen). Ein Soundambiente Generator kann beim Donnern eines Blitzes einen Blitzeffekt starten. Eine Software welche auf Miele@Home reagiert kann visuell mitteilen dass die Waschmaschine fertig ist. Eine Kalendersoftware hat die Möglichkeit an einen Termin zu erinnern usw.

Allgemein ist die PC Steuerung weit flexibler als es jede Microcontrollersteuerung sein kann, da sie um einiges leichter erweiterbar ist. Zusätzlich bietet der PC weitere Möglichkeiten wie Zugriff auf eine Musikdatenbank um so auch gleichzeitig als Stereoanlage oder Internet Radio zu arbeiten.

Auch wie angesprochen kann ein dynamischer Ambiente Generator auf dem PC betrieben werden, der z.b. passend zum Ambiente (Lagerfeuer im Wald, Unwetter in der Nacht, Sonnenaufgang am Strand, Dungeon, ...) dynamisch eine Soundkulisse generiert und dazu passend ein Lichtsetting startet bzw. Effekte abspielt.

Auch wenn viele dieser Ideen als Spielereien wirken mögen, so geht doch der Trend in genau diese Richtung. Erste Ansätze wie Philips AmbiLight gehen in diese Richtung, sind jedoch noch nicht flexibel genug.

Hardware

Die eigentlichen Controller Platinen sind das was NICHT inhouse geschehen kann.

Lichtmodule

In dieser Beispielininstallation existieren aktuell 28 Lichtmodule. Gesteuert werden sie über 14 Controllerplatinen (jeweils 2 pro Platine).

Ein Lichtmodul wird voraussichtlich aus den folgenden Elementen bestehen :

http://www.leds.de/p103/LED_Leisten_Module/SuperFlux_LED_Leisten/SuperFlux-LED_Leiste_rot_30x2000mcd_120_500mm_12V.html

http://www.leds.de/p102/LED_Leisten_Module/SuperFlux_LED_Leisten/SuperFlux-LED_Leiste_gruen_30x6000mcd_90_500mm_12V.html

http://www.leds.de/p101/LED_Leisten_Module/SuperFlux_LED_Leisten/SuperFlux-LED_Leiste_blau_30x3000mcd_90_500mm_12V.html

Dabei wird ein Modul aus 3 roten, 1 grünen und zwei blauen Leiste bestehen. (Alternativ 1-1-1).

Die Spannungsversorgung der Lichtmodule beträgt 12 V. Eine passende Stromversorgung ist damit also (z.b. spezielle LED Traffos, PC Netzteile) nicht allzu aufwendig und muss nicht extra entwickelt werden.

Die einzelnen Farben der Lichtmodule müssen über PWM (mit 100hz) gesteuert werden.

Controller platine

Jede Controllerplatine besitzt einen Microcontroller, 6 Leistungstransistoren für die PWM Steuerung, eine Dipschalter Adresskonfiguration (mindestens 5Bit für bis zu 32 Platinen -> maximal 64 Module), sowie einen Ein/Ausgang für den Datenbus.

Die Controllerplatine besitzt eine eigene Stromversorgung welche von dem Stromkreis der LED's möglichst getrennt verläuft.

Da für den Bus primär eine Ringtopologie anstelle einer Sterntopologie gewünscht ist, braucht die Platine einen Buseingang und einen Ausgang. Sollte es technisch (Ein/Ausgänge des Controllers) möglich sein, so ist die Frage ob der Ausgang anstelle das Signal nur elektrisch durchzuschleifen, aktiv gesteuert wird (d.h. Der Controller sendet alle Signale die er bekommt auch gleich weiter und verstärkt bzw. säubert sie so).

Ob der Bus Glasfaser, Kabel oder Funkbasiert ist, ist in diesem Zusammenhang bedeutungslos. Gewünscht ist nur dass er möglichst billig zu realisieren ist und mit Standardkomponenten bei Kabeln auskommt.

Jeder Controller reagiert auf zwei Adressen. (Gerade und Ungerade)

Beispiel :

eingestellte Adresse : 11111

Controller reagiert auf : 111110 und 111111

Die niedere Adresse steuert Lichtmodul 1 (jeweils RGB), die höhere Lichtmodul 2

Controller Firmware

Die Firmware des Controllers kann ziemlich simpel sein da die Hauptsteuerung ja der PC übernimmt.

Somit muss die Firmware (in der Minimalausführung) nicht mehr machen als die Signale des Busses abzugreifen, für sich die Helligkeitsinformationen zu holen und seine PWM Ausgänge zu steuern.

Eine Speicherung verschiedener Farben oder Lichtmuster sind nicht notwendig.

Details zur PWM Steuerung

Die PWM Steuerung muss mindestens 100Hz betragen.

Wenn möglich sollte sie mit 10 Bit gesteuert werden können, 8 Bit sind aber ausreichend falls sich 10 Bit als zu kostenintensiv gestaltet. Der Grund für die 10 Bit Steuerung ist die Möglichkeit hierbei die Nichtlinearität der Farbcharakteristik zu koeffizieren (eine 8 Bit Farbe wird auf dem PC in einen 10 Bit Wert gewandelt anhand einer dort abgelegten Kalibrierungskurve).

Animierbare Helligkeit

Der Controller bekommt pro Ausgang einen Ziel Helligkeitswert und einen Zeitpunkt zu dem er diesen erreichen muss sowie ob die Helligkeit interpoliert oder abrupt umgeschaltet wird.

Dazu ist es notwendig dass alle Controller die selbe Zeit haben. Um dies zu ermöglichen gibt es ein spezielles Init Commando dass den Zeitcounter ALLER Controller auf 0 zurücksetzt.

SteuerCommandos

Über den Bus sollten folgende Kommandos kommen.

ResetTimer

setzt die Zeiten aller Controller zu 0. Die Farbkommandos werden NICHT zurückgesetzt. (das heisst das letzte Farbkommando bleibt erhalten)

setColorInterpolated <Adresse 8Bit> <RGB> <Zeit>

Zur Adresse siehe Controllerplatine (jeder Controller hat 2 Adressen).

Die Daten sollten wenn möglich in 10 Bit Genauigkeit übertragen werden. Sollte dies aber zu aufwendig sein, so sind auch 8 Bit möglich (siehe PWM)

Die Zeit definiert bis zu welchem Zeitpunkt die eingestellte Helligkeit erreicht werden soll.
SetColor <Adresse 8Bit> <RGB> <Zeit>

Wie setColorInterpolated jedoch wird hier die Farbe abrupt zum Zeitpunkt X gewechselt.

Der Controller sollte die Möglichkeit haben 2 Farbkommandos pro Adresse zu speichern (aktuelles und nächstes Farbkommando). Dies um zu vermeiden dass durch Zeitversatz „Ruckler“ in der Beleuchtung entstehen.

Die einzelnen Farbkommandos können dabei überschrieben werden.

(Beispiel : Kommando1, Kommando2 sind beide noch nicht ausgeführt. Wird ein drittes Kommando übertragen, so wird das zweite Kommando überschrieben).

Sonstiges

Ziel des Projektes ist es zum einen möglichst erweiterbar zu sein und zum anderen einfach zu montieren. Das heisst bei der Verkabelung sollte auf einfache Verbindungen geachtet werden.

Die LED Leisten sind nur als Beispiel gedacht. Für andere Projekte können auch stärkere (oder schwächere) Leistungen kommen.