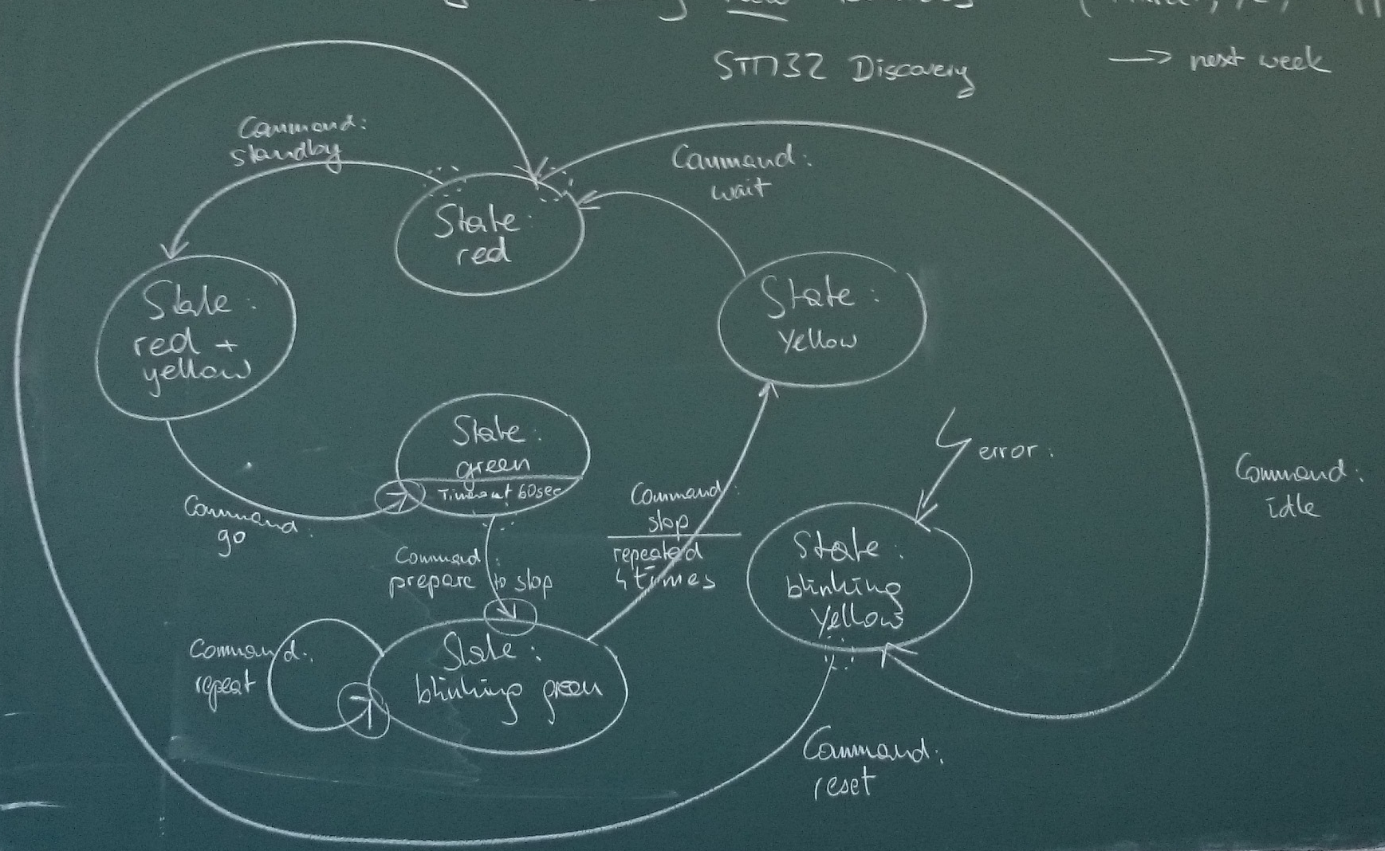
IndInf Systemtechniklabor – Ampelbeispiel

# Aufgabenstellung

Implement a component based C-Programm to show the difference of the 5 types of state machines presented in the book of Mrs. Elicia White "Making Embedded Systems" with traffic light system we discussed in the lesson. To test your implementation you can use simple output functions (e.g. fprintf), but be prepared to implement it also on hardware (GPIO with Leds, Timers, etc.).

Don't forget to document the differences (advantages/disadvantages) in your protocol.



# Vorbereitung

Um die Applikationen zu testen und zu auf den STM zu flashen wird mit eclipse gearbeitet.

Die Website <http://www.openstm32.org> bietet ein Tutorial zur korrekten Konfiguration der IDE.

Damit mit der HAL-Library gearbeitet werden kann, muss eine eclipse Erweiterung über „Install new software“ von der Website

<http://www.ac6-tools.com/Eclipse-updates/org.openstm32.system-workbench.site> geladen werden.

Damit der STM unter Windows erkannt wird und für das spätere flashen bereitsteht, muss ein Treiber von der Website <http://www.st.com/web/catalog/tools/FM147/SC1887/PF260218> geladen und installiert werden

# Erstellen des eclipse Projekts

Beim Wizard müssen die Einstellungen der neu installierten Software benutzt und die STM Einstellungen angepasst werden (Model, …). Dafür wird die Toolchain Ac6 STM32 MCU GCC verwendet.

Nun kann die spezielle Debug-Configuration (AC6) verwendet werden um das Projekt zu debuggen.

In manchen Fällen kann es notwendig sein, dass der STM an einem USB2.0 Port angeschlossen ist, und eclipse als Administrator ausgeführt wird.

# Vor- und Nachteile der State-Machines

## State-Centric State Machine

In einem Switch-Case Konstrukt wird der derzeitige Status der Ampel abgefragt (Rot, Gelb, …). Wenn das zugehörige Event stimmt, nimmt die Ampel den nächsten Status an.

### Vorteile

* Relativ einfach zu Verstehen und umzusetzen

### Nachteile

* Komplett statisch
* Jeder Zustand muss über die anderen Bescheid wissen
* Switch Konstrukt wird sehr verschachtelt wenn viele Zustände vorhanden sind

## State-Centric State Machine with hidden Transitions

Funktioniert vom Prinzip her wie die State-Centic Version, verzichtet allerding komplett auf die Abfrage der Events. (z.B. Wenn der Status Rot ist wird er auf Rot-Gelb geändert)

### Vorteile

* Vereinfachte Version der State-Centric Version
* Bessere Verkapselung
* Geringere Abhängigkeiten

### Nachteile

* Ebenfalls komplett statisch

## Event-Centric State Machine

State-Pattern State Machine

Table-Driven State Machine

# Quellen

[1] OpenSTM32 Community Site - <http://www.openstm32.org/>

zuletzt abgerufen am 25.09.2015 (online)