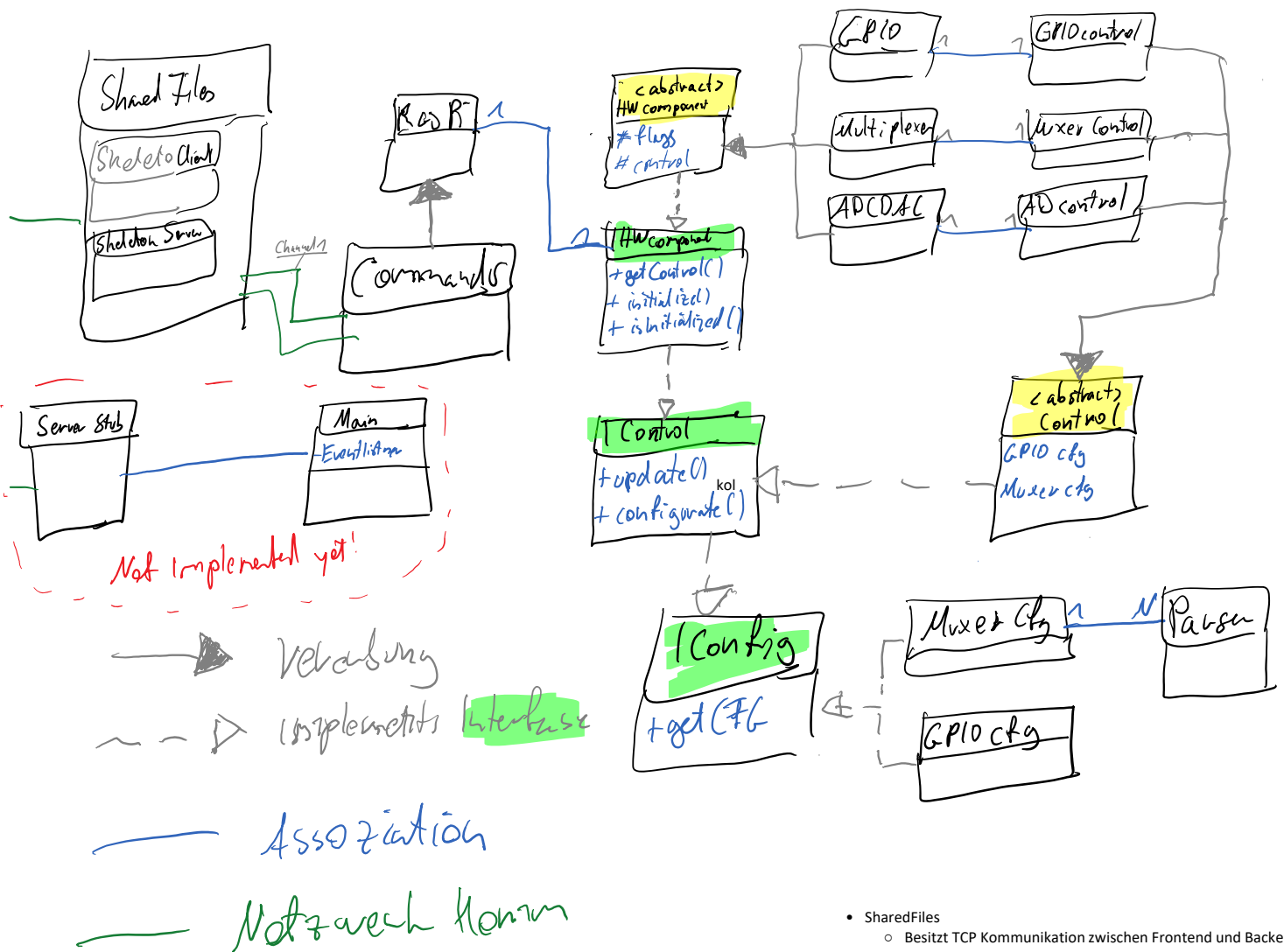


LCD: control_turbchlauf() AMOS()



Änderungen im Arbeitsablauf

- Commands erstellen / ändern
 - In Klasse Commands ändern/ hinzufügen
 - Aufruf Basisbefehle mittels: z.B. : LCD.Control.toggleBacklight()
- Basisbefehl erstellen / ändern
 - In Klasse welche IControl implementiert erstellen / ändern
- Konfiguration von Pin Belegung ändern
 - Unterklasse von IConfig anpassen

- SharedFiles
 - Besitzt TCP Kommunikation zwischen Frontend und Backend
- RasPi
 - Initialisiert Hardware und start-up Konfiguration des Raspberry Pis
 - Besitzt Instanzvariablen aller Hardware Komponenten (Könnte zu Commands geschoben werden)
- Iconfig
 - Interface welches verwendet wird um auf Konfigurationsinformationen zu zugreifen z.B. für Multiplexer und GPIO Pin Belegung. Eine Auslagerung der Basis GPIO Konfiguration wäre hier auch denkbar.
- Icontrol
 - Interface welches verwendet wird um allgemeine Manipulationsmethoden auf der Hardware aufzurufen z.B. eine Konfiguration auszuführen.
- IHWcomponent
 - Interface zu Interaction mit Hardware Komponenten. Abfrage nach Initialisierung, Control Schnittstelle holen etc.
- Abstract Control
 - Besitzt aktuelle Konfigurationsinformation (Gpio, Muxer)
 - Zentrale Methode zur configuration der HI pin belegung (Muxer)
- Abstract HWComponent
 - Besitzt initialisierungsinformation
 - Besitzt control Schnittstelle der Hardware componente
- Commands
 - Besitzt Methoden zur Kombination von Basisbefehlen.
 - Nutzt Felder von RasPi die mit Hwcomponenten initialisiert sind
- Parser
 - Liest z.B. xml files aus, oder generiert sie
- Main
 - Beobachtet Events auf dem Backend und sendet sie weiter an den ServerStub

- ServerStub
 - Gibt events weiter an das Frontend