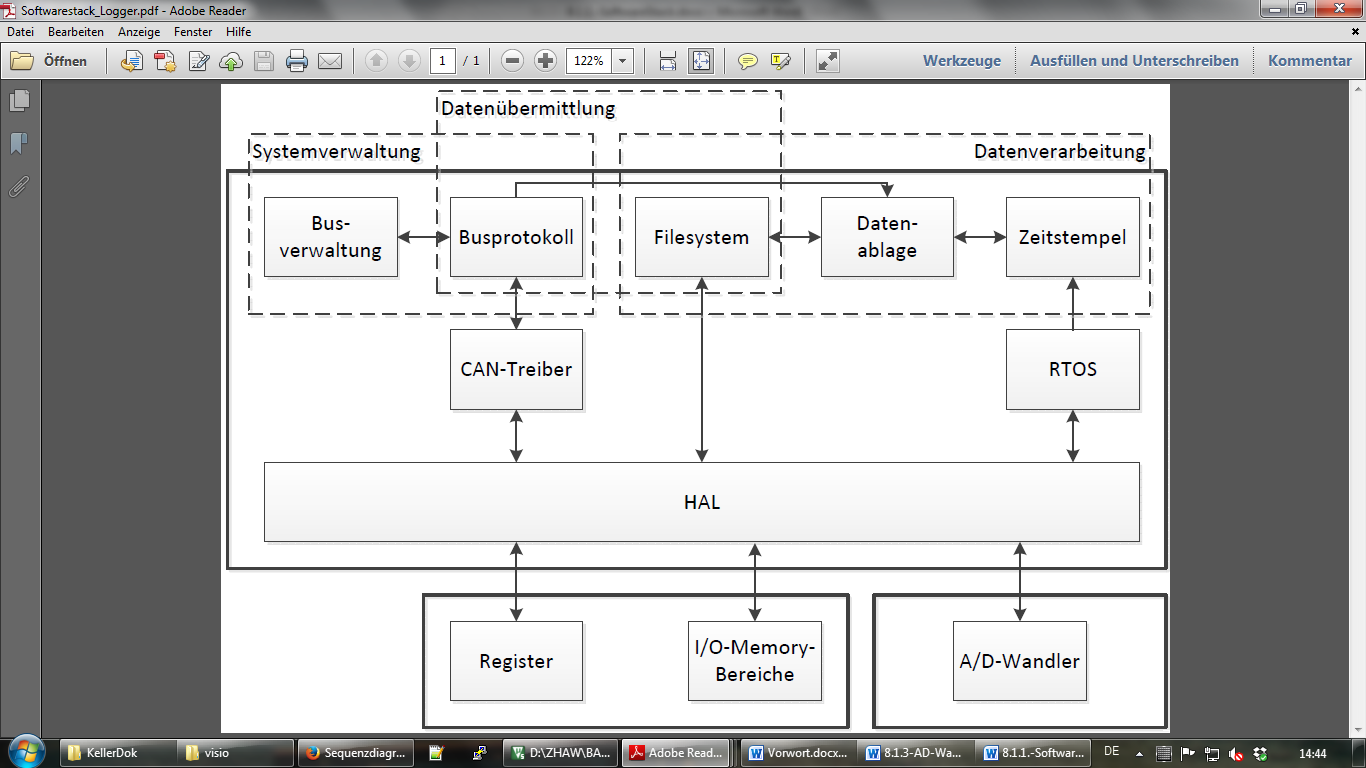
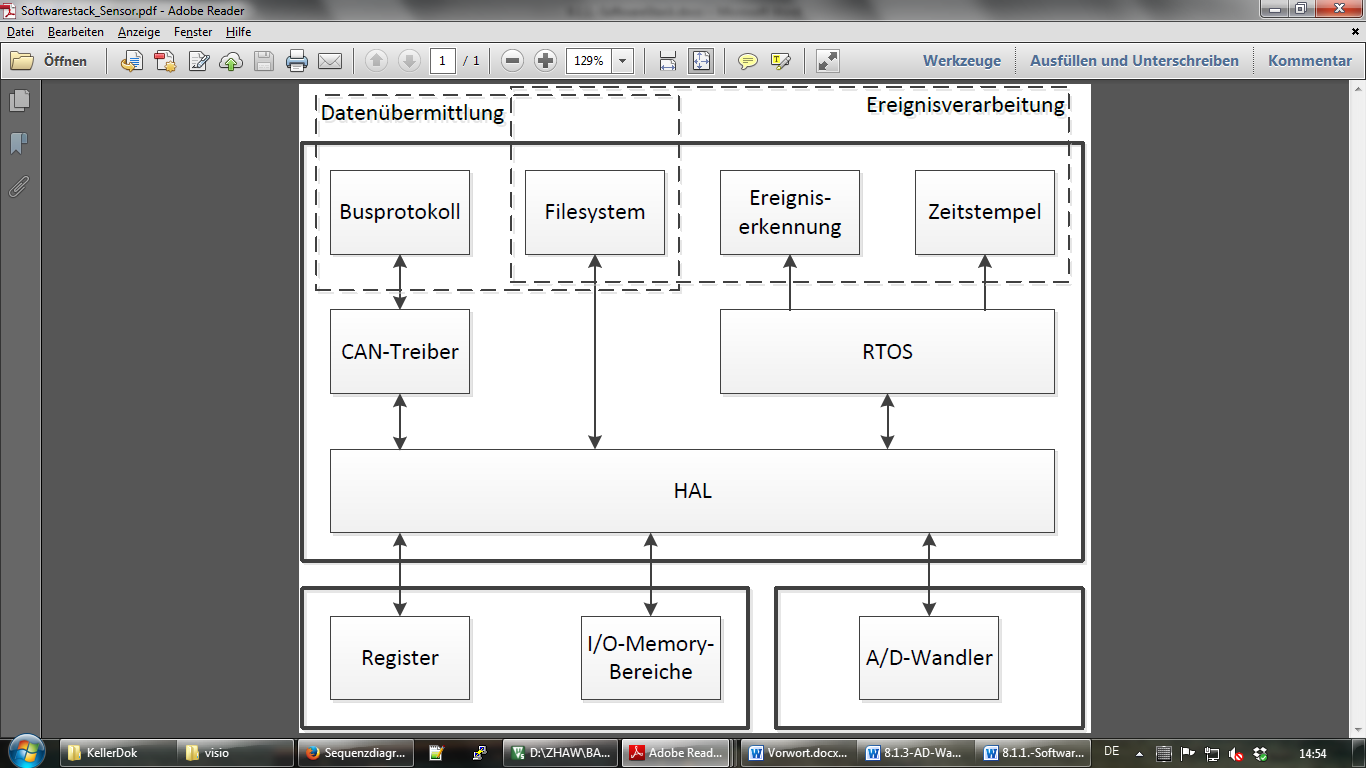
8.1.1. Überblick

Logger



Die Hauptkomponenten des Datenloggers sind die Datenablage und das Busprotokoll, welche jeweils in einem eigenen Thread implementiert wurden, damit sie unabhängig voneinander arbeiten können. Damit die Kommunikation zwischen den Teilnehmern reibungslos funktioniert wurde eine Busverwaltung implementiert, die mehrere Aufgaben wahrnimmt. Zum einen kümmert sie sich um die Teilnehmerverwaltung und verteilt, basierend auf einem Konfigurationsfile und der Seriennummer der Sensoren, eindeutige Identifier, welche zur Kommunikation unabdingbar sind. Weiter weist die Busverwaltung auch einem Teilnehmer ein Token zu, wenn dieser Daten schicken darf. Damit wird sichergestellt, dass kein Sensor verhungert, weil seine Nachrichten jeweils von anderen übersteuert werden. Das Busprotokoll kümmert sich dabei um die technische Kommunikation mit den Sensoren am Bus und bietet Schnittstellen an, um Nachrichten abzuschicken oder eingetroffene Meldungen abzuholen. Trifft eine Meldung von einem Sensor ein, wird ein Interrupt ausgelöst, der die Meldung ausliest und in einem Ringbuffer ablegt. Dieser Buffer wird periodisch vom Kommunikationsthread ausgelesen und die darauf enthaltenen Meldungen in ein Format gebracht, das für die Anwendung lesbar ist. Im Anschluss wird diese Meldung mittels einer Queue und unter Verwendung eines Memory-Pools dem Datenablage-Thread zur Verfügung gestellt. Dieser holt sich die vorbereitete Meldung und, sofern es sich dabei um Messdaten handelt, schreibt diese in das Logfile des sendenden Sensors. Nach der Verarbeitung der Message gibt der Datenablage-Thread den Platz im Memory-Pool wieder frei. Bei ausgehenden Nachrichten wird gleich verfahren, auch hier schreibt der Absender die Meldung in eine Queue, die dann vom Busprotokoll-Thread abgearbeitet wird. Beim Logger wurde die Queue für die eingehenden Meldungen gross genug gewählt, um ca. 800 Meldungen zwischen zu speichern, falls viele Ereignisse erfasst und übermittelt wurden. Dem gegenüber wurde die Queue für die ausgehenden Meldungen klein gehalten, da der Logger nur Steuerkommandos oder Konfigurationen an die Sensoren übermittelt, was nicht so häufig vorkommt, bzw. schnell erledigt ist.

Sensor



Der Sensor ist ähnlich dem Logger strukturiert, auch er operiert über zwei Threads. Zum einen ist hier auch wieder der Busprotokoll-Thread anzutreffen, der wie beim Logger auch die ganze Kommunikation kapselt. Im zweiten Thread läuft auf dem Sensor die Ereigniserkennung, welche die Daten des angeschlossenen Beschleunigungssensors ausliest und die Ergebnisse, basierend auf dem festgelegten Betriebsmodus aufbereitet. Ist ein Ereignis abgeschlossen (d.h. der Ausschlag des Beschleunigungssensors liegt unter einem festgelegten Schwellwerts) wird dieses in die Queue zum Busprotokoll geschrieben. Hat der Sensor das Token bereits erhalten, wird er die Meldung unverzüglich übermitteln, andernfalls wird er die Message in der Queue belassen, bis ihm das Token zugewiesen wird. Damit die Queue nicht so schnell überläuft, wurde sie entsprechend gross gewählt und dient somit als eine Art Buffer. Die Queue für eingehende Meldungen bietet dafür nur wenigen Meldungen Platz, da die Sensoren einerseits nur Meldungen erhalten, die entweder an sie selbst oder als Broadcast gesendet wurde (andere Meldungen werden vom Acceptance-Filter ausgeschlossen) und andererseits nur Steuerkommandos vom Logger zu erwarten sind.