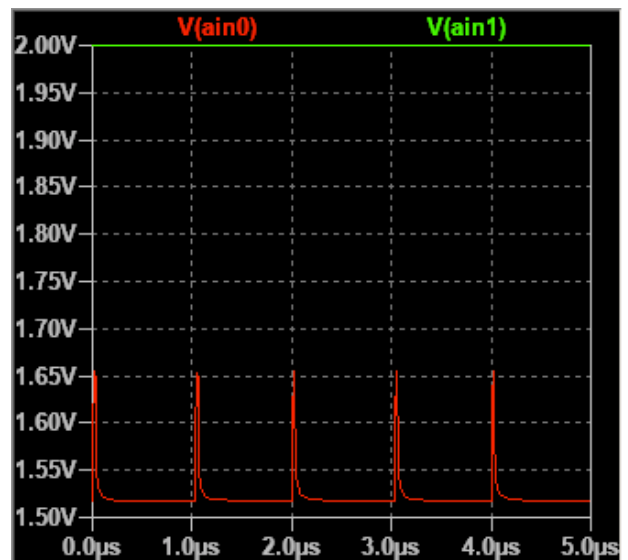
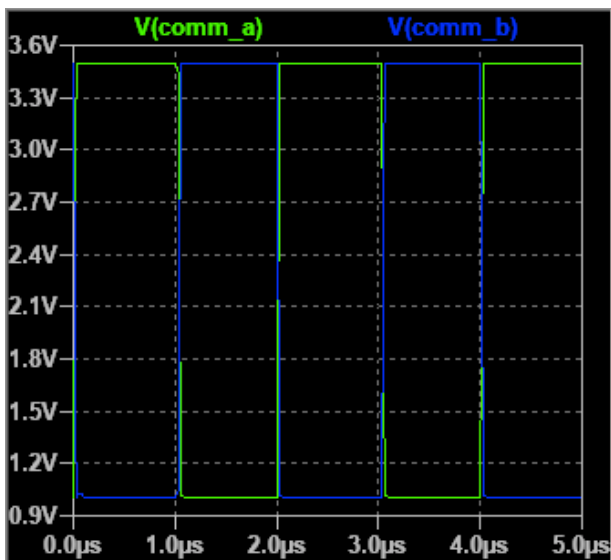


## Programm allgemein

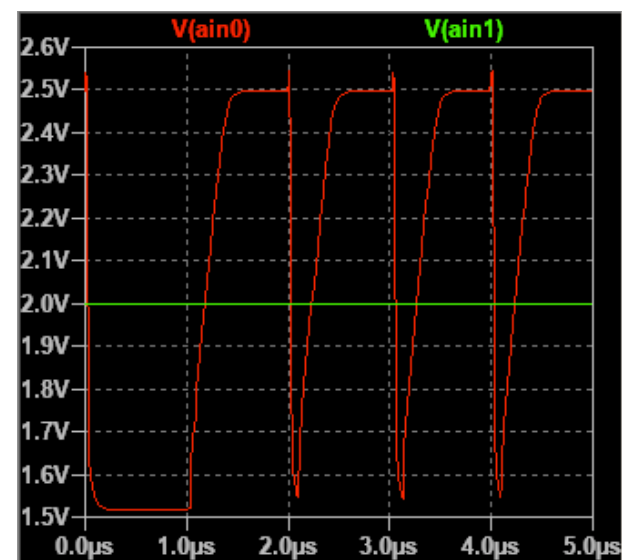
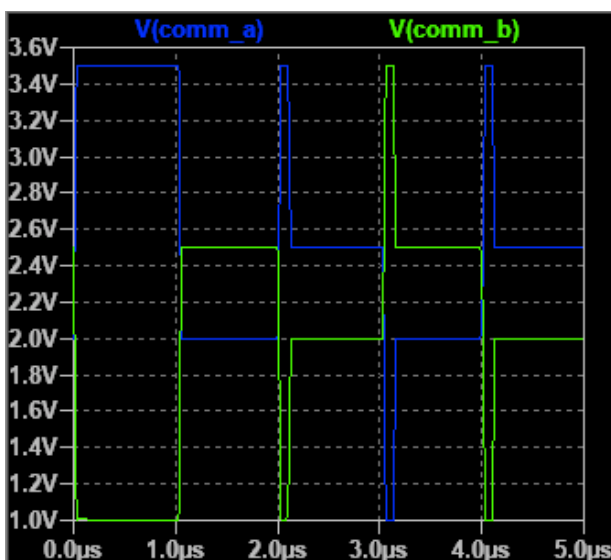
- Aufgaben:
  - System Timer mit Timer1 (16 bit) -> 1us bis 1ms Systemtakt für Buszugriff, Debounce, Systemzeit und Timings mit geringer Priorität  
(muss man testen, wie tief man mit der Zeit gehen kann)
  - PWM Steuerung der RGB LED -> Hybrid PWM (Timer 0 Interrupts + Software)
  - Button Auswertung und Debouncing, ohne Interrupt.
  - INT0 (Button) und INT1 (Bus) nur zum wecken aus IDLE verwenden.
  - Buszugriff mit System Timer und UART
  - IR LED Datenübertragung Timing mit System Timer
- Adressierung
  - Jeder Controller bekommt den seine laufende Nummer (oder einen Hash davon, oder eine eindeutige Zufallszahl) in den EEPROM geschrieben. Dieser Wert dient zur Adressierung des Controllers auf dem Bus.
  - Um der Steuereinheit die Adressen und Positionen bekannt zu machen muss eine Setup Prozedur entwickelt werden. Z.B. kann von der Steuereinheit ein Setup Modus ausgerufen (Broadcast) werden. In diesem sendet jede LED Box bei Tastendruck ihre eigene Adresse an die Steuereinheit. Um die Positionen zuzuordnen verlangt die Setup Prozedur eine feste Reihenfolge in der die Taster gedrückt werden müssen (z.B. A1 bis A12 und dann B1 bis B12).
- Zufall
  - Die Adresse dient zusätzlich als erster seed für den Zufallsgenerator, damit von Anfang an verschiedene Zufallszahlen garantiert werden können
  - Bei jedem Tastendruck wird die aktuelle Systemzeit, evtl. verrechnet mit der Adresse, als neuer seed verwendet.

## Buszugriff

- bisher geplant: CSMA/CD
  - Funktioniert mit SN75176B nicht, da:
    - Hysterese im Empfänger zu klein oder Strombergrenzung des Senders zu hoch (max. 250mA)
    - Das führt dazu, dass aufgrund der Leitungslänge nicht sichergestellt ist, dass Kollisionen erkannt werden. Außerdem ist der Strom recht hoch.
- Neuer Vorschlag: CSMA/CA
  - Kollisionen werden von vornherein vermieden.
  - Für den unwahrscheinlichen Fall einer Kollision: Kollisionsdetektor
    - Wenn der Low Pegel auf dem Bus rund 1.4V überschreitet liegt eine Kollision vor.  
Da der Low Level Treiber schwächer ist als der High Level Treiber wird dieser Pegel ausgewertet (vgl. DS Fig.1 und Fig.2).
  - Simulation mit gleichstarken Treibern (immer, wenn  $A_{IN0} > A_{IN1}$  wird eine Kollision detektiert)
  - Normalzustand



- Kollision



- Ablauf:

1. Teilnehmer wird eingeschaltet
2. Teilnehmer wartet für mindestens die CA (Collision Avoidance) Periode oder die Sendedauer eines Bytes (welche länger ist) und beobachtet den Bus
3. Wenn der Bus frei ist (RX = High Pegel) kann der Teilnehmer senden.  
Dazu wird der Bus auf Low Pegel gezogen, es beginnt die CA Periode.
4. Nach einer zufälligen Zeit in der CA Periode wird der Bus freigegeben (Sender auf tristate)
5. Wenn der Bus auf Low bleibt kann nicht gesendet werden, da ein anderer Teilnehmer senden will und zufällig die längere Zeit „gewürfelt“ hat. Weiter mit 8.
6. Wenn der Bus auf High springt kann nach einem kurzen Delay (alle Teilnehmer müssen nach der Low-High Flanke aufwachen) angefangen werden zu senden.  
Hierfür wird die UART Peripherie benutzt. Zuerst wird die Adresse des Empfängers gesendet. Danach können alle nicht angesprochenen Teilnehmer wieder schlafen.
7. Sollten zwei Teilnehmer gleichzeitig angefangen haben und auch die gleiche Zeit gezogen haben wird jetzt die Kollisionserkennung anschlagen. In diesem Fall muss von dem Teilnehmer, der sendet und die Kollision erkannt hat, die Kollisionserkennung der anderen Sender provoziert werden. Dafür wird der Bus für die Sendedauer eines Bytes Low gehalten.
8. Wenn der Bus länger als die Sendedauer eines Bytes High bleibt ist er wieder frei.  
Zurück zu 3.

- warum nicht Daisychain?

- Für Daisychain würde ein weiterer Kommunikationsport am Mikrocontroller und ein weiterer Transceiver benötigt. KO Kriterium ist hier der nicht verfügbare weitere Kommunikationsport.
- Außerdem müsste jedes mal die Nachricht über alle Zwischenstationen gereicht werden. Bei Fehler einer Station kann ein ganzer Bereich nicht mehr angesprochen werden.