

Arduino Day 2017
Hochschule Niederrhein
UART Smartwatch - Prototyp und Android App

Jochen Peters

<http://dummer.click>

April 2017

- 1 UART Smartwatch
- 2 Low Cost vs teuer
- 3 Akku Laufzeit
- 4 Mini Beispiel
- 5 Farbdisplay

Die Idee

- Handy in Tasche pipt - Wichtig?
- Nachrichten und Zeit vom Handy
- UART Bluetooth
- OLED Display
- DEINE EIGENEN MODS!!

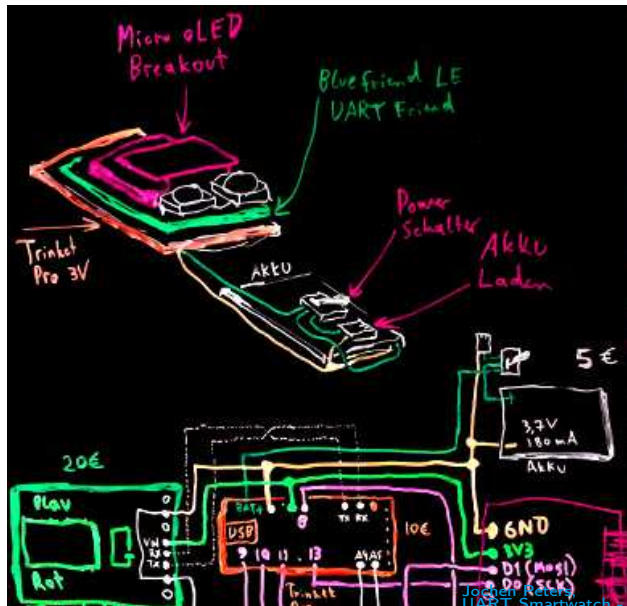
Bauteile

Erster Prototyp teuer (deutsche Händler):

- Bluefruit LE UART (19 EUR)
- SparkFun Micro OLED (17 EUR)
- Adafruit Pro Trinket 3V, 12MHz (10 EUR)
- Lipo 3,7V - 180mAh (6 EUR)

ergibt 52 EUR :-(

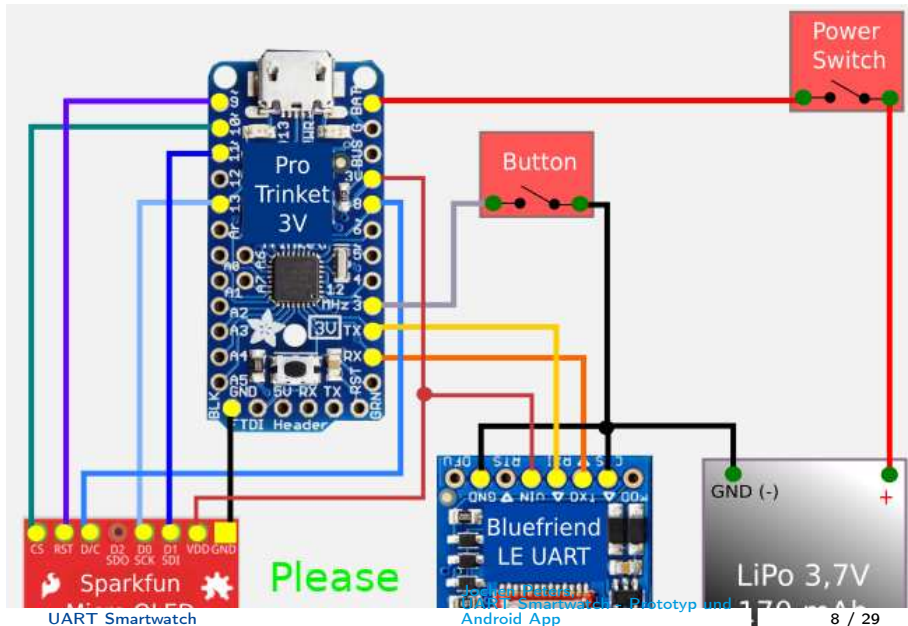
teurer Prototyp



UART Smartwatch

Jochen Meyers
UART Smartwatch - Prototyp und
Android App

teurer Prototyp

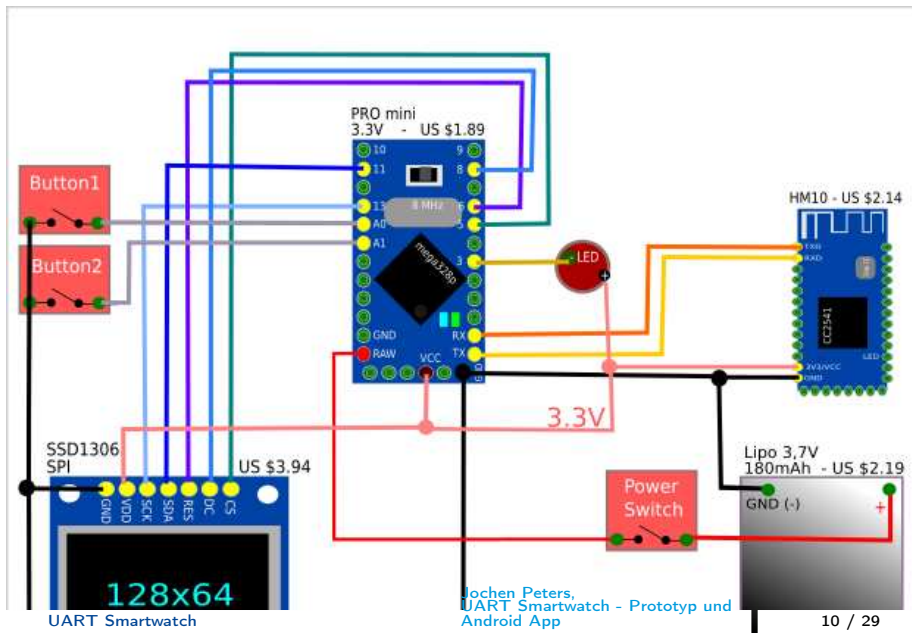


Bauteile Low Cost (Asien):

- Bluetooth HM-10 (2,60 EUR)
- ssd1306 SPI OLED (2,80 EUR)
- Pro Mini 3V 8MHz (1,60 EUR)
- Lipo 3,7V - 180mAh (5,50 EUR)

ergibt ca 13 EUR :-D

billige Version



billige Version



Low Cost vs teuer

- Display mit 128x64 statt 64x48 ist billiger (braucht mehr RAM)
- Programmieren nur mit Zusatz Hardware
- bis zu 60 Tage Lieferzeit
- Bluefruit = 3mA , HM10 Modul = 10mA

Fazit: Kompromisse finden. Dumm, wenn Display aus und Arduino im Sleep Modus trotzdem Akku keine 8h durchhält. In der teuren Version: 30h !!!

Akku Laufzeit

Sleep Modus des Arduino Mini brachte 30h, aber:

- 1 Uhr zählte Sekunden nicht selbst
- 2 Uhr musste immer Zeit holen
- 3 Wann ist Akku wirklich leer? Abnutzung?
- 4 keine LED Benachrichtigung bei neuen Nachrichten
- 5 Mit 3 EUR mehr: besseres Bluetooth Modul?

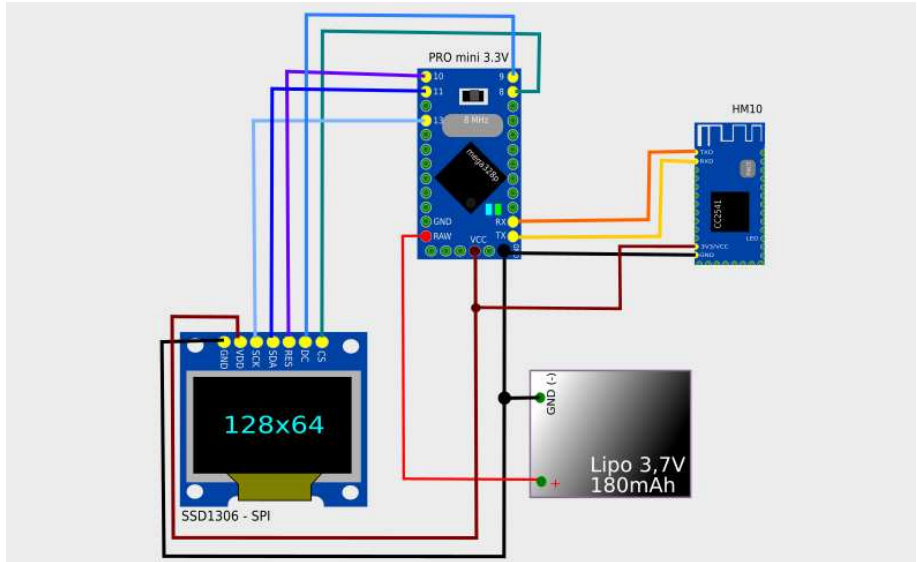
zu 4: Mit UART Arduino aufwachen lassen - Der Sleep Modus bringt nicht viel

github.com/no-go/Bluetooth-Smartwatch

Mini Beispiel

no-go.github.io/Android-nRF-UART

Mini Beispiel - Schaltung



Mini Beispiel - Firmware

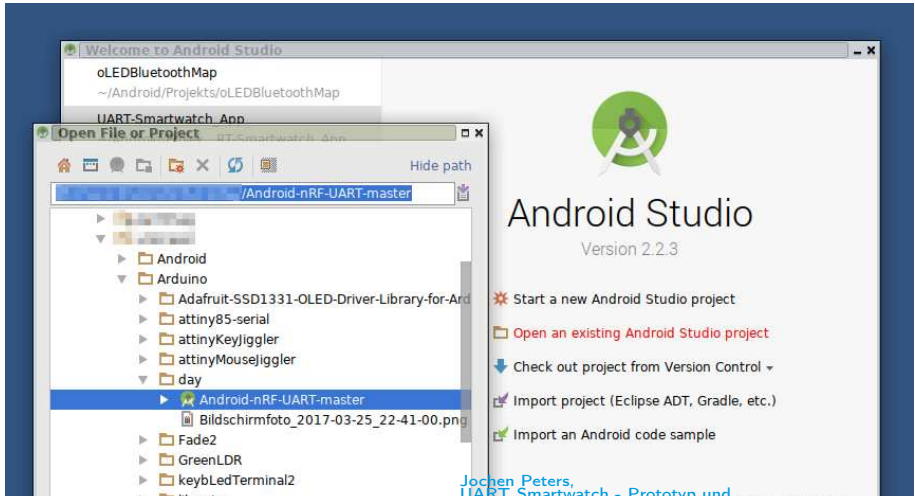
```
1 #include <Adafruit_GFX.h>
2 #include <Adafruit_SSD1306.h>
3
4 // OLED (11 -> MOSI/DIN, 13 -> SCK)
5 #define PIN_CS      8
6 #define PIN_RESET  10
7 #define PIN_DC      9
8 Adafruit_SSD1306 oled(PIN_DC, PIN_RESET, PIN_CS);
9
10 #define BUFLen 168
11 int count = 0;
12
13 void setup() {
14   Serial.begin(9600);
15   oled.begin();
16   oled.setTextColor(WHITE);
17   oled.setTextSize(1);
18   oled.clearDisplay();
19   oled.display();
20 }
21
22 void loop() {
23   while (Serial.available()) {
24     if(count >= BUFLen) {
25       oled.clearDisplay();
26       oled.setCursor(0,0);
27       count = 0;
28       oled.display();
29     }
30     char inChar = (char)Serial.read();
31     oled.print(inChar);
32     oled.display();
33     count++;
34   }
35 }
```

Mini Beispiel - Android App (1)

- Mit Android Studio kostenlos selber Apps machen
- Nordic Semiconductor: UART App
- viel OpenSource + Beispiel
- ein *rooten* ist nicht nötig

Mini Beispiel - Android App (2)

Download der UART Terminal App von github
github.com/NordicSemiconductor/Android-nRF-UART
und öffnen in Android Studio :-D



Mini Beispiel - Android App (3)

Änderungen der App

- Eingabefeld von 20 auf 168 Zeichen vergrößern
- Schleife, in 20 Zeichen Intervallen zu senden
- Anpassen auf *billig Bluetooth Modul*
- Benachrichtigungs Diensts erstellen
- Nachrichten durch App/Activity senden lassen

Mini Beispiel - Android App (5)

Von mir detailliert beschrieben mit Screenshots auf:
no-go.github.io/Android-nRF-UART

Farbdisplay

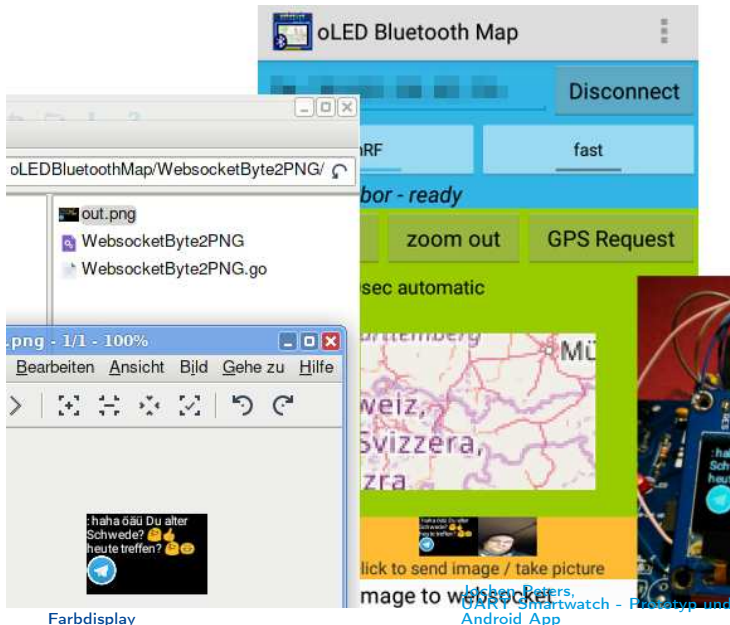
SSD1331 nicht viel anders als SSD1306

- Anschluss identisch
- langsamer (16 statt 1 Bit pro Pixel)
- teurer (min 10 EUR statt 3 EUR)
- Icons, Bilder, Emoticons möglich

Farbdisplay Experimente



Farbdisplay Experimente



- App und Firmware github.com/no-go/UART-Smartwatch/tree/gplay
- mini Beispiel no-go.github.io/Android-nRF-UART
- Artikel [vibora.de vibora.de/2016/04/do-it-yourself-smartwatch.html](http://vibora.de/vibora.de/2016/04/do-it-yourself-smartwatch.html)
- Farbdisplay Arduino
github.com/no-go/UART-Smartwatch/tree/ssd1331
- oLED Bluetooth Map no-go.github.io/oLEDBluetoothMap

Danke

Fragen?