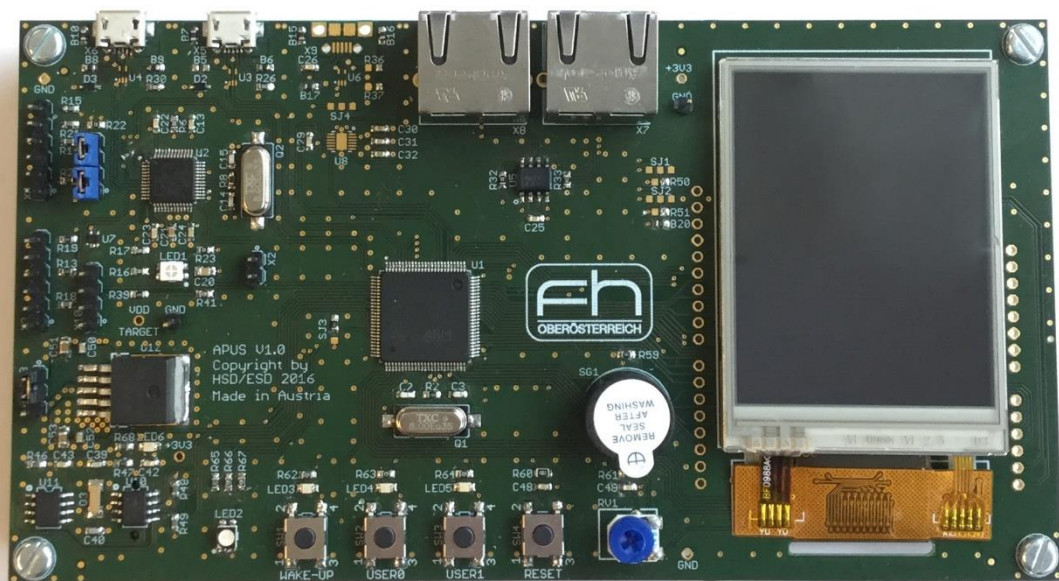




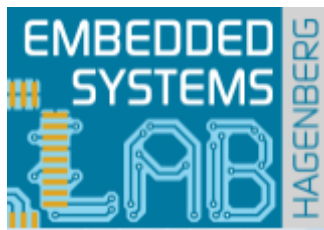
Entwicklungsboard



Handbuch

Version 1.0

HARDWARE-SOFTWARE-DESIGN



FH OÖ Campus Hagenberg
Studiengang Hardware Software Design
Forschungsgruppe Embedded Systems
Softwarepark 11
A-4232 Hagenberg

E: hsd@fh-hagenberg.at

T: +43 50804 22400

F: +43 50804 22499

Autor: FH-Prof. Dr. Josef Langer

Inhaltsverzeichnis

Einleitung.....	4
Der Name APUS.....	4
Übersicht APUS.....	5
Bestandteile:.....	5
Die wichtigsten Funktionen von APUS	5
Die ersten Schritte	7
Funktionen des Testprogramms.....	7
Bildschirm	8
Leuchtdioden.....	8
Potentiometer	9
Buzzer	9
Temperatursensor.....	10
USB Mausemulator.....	10
Zeichenprogramm	11
CAN Test	11
Real Time Clock (RTC)	12
Aufbau APUS	13
Spannungsversorgung	13
Mikrocontroller STM32F072	14
User Interface	15
Temperatursensor und RTC Real Time Clock	16
Jumper Konfigurationen und Test-Pins	17
Auswahl der Spannungsversorgung	17
Konfiguration der Programmierschnittstelle	17
ST-Link/V2-A Virtual COM	18
Spannungsmesspunkte.....	19
Arduino Shield	20
Arduino WiFi/Bluetooth Shield	20
Arduino Sensor Shield	21
Arduino Ethernet Shield	22

Einleitung

Das Entwicklungsboard APUS wurde für den Studiengang Hardware-Software-Design zum Zwecke einer praxisorientierten Lehre entwickelt. APUS ist speziell abgestimmt auf die Lehrinhalte Mikroprozessortechnik, hardwarenahe Programmierung und Kommunikationsnetze sowie für Embedded Operating Systems im Masterstudiengang Embedded Systems Design.

Für das Entwicklungsboard APUS stehen Schaltpläne, Datenblätter und Dokumentationen zur Verfügung. Ein Board Support Package (BSP) und ein Demoprogramm garantieren einen einfachen Einstieg und die schnelle Entwicklung von Projekten. Diese Softwarepakete werden laufend aktualisiert und erweitert.

Der Name APUS

Der Name APUS stammt aus dem Lateinischen und bedeutet Paradiesvogel. APUS ist zugleich ein Sternbild, welches einen tropischen Vogel darstellt. Die Freiheit, Leichtigkeit und Buntheit des Paradiesvogels soll Motto für Innovation, Kreativität und Ideenreichtum bei der Entwicklung von Anwendungen mit APUS sein.



Abbildung 1 Logo APUS

Übersicht APUS

Bestandteile:

Das Development-Kit besteht aus:

- einem Entwicklungsboard,
- zwei USB-Kabel und
- einer Trage-Tasche



Abbildung 2 APUS Board, Box und Kabel

Die wichtigsten Funktionen von APUS

APUS bietet folgende Funktionen:

- Real Time Clock (RTC) mit SRAM und EEPROM (I²C)
- Temperatur Sensor (I²C)
- USB-Funktion
- UART über USB (VCOM) oder RS232
- 2,4" TFT Touch LCD (240x320 Pixel, 16 Bit Farben)
- 2 CAN Bus Schnittstellen
- USB-Programmier- und Debuginterface
- Arduino Shield Steckplatz
 - Erweiterung für TCP/IP
 - Erweiterung für Bluetooth / Sensorik
- 1 RGB-LED, 3 LEDs (rot, grün, orange)
- 3 Taster
- 1 Potentiometer
- 1 Buzzer

APUS kann über jede der drei verfügbaren USB-Buchsen mit Strom versorgt werden. Die USB-Buchsen haben folgende Funktionen:

- USB (1): Programmier- und Debugport
- USB (2): USB-Function für beliebige Anwendungen
- USB (3): UART mit VCOM-Implementierung

Die CAN Schnittstelle (4) bzw. (5) sind beide mit der CAN Peripherie des Mikrocontrollers verbunden.

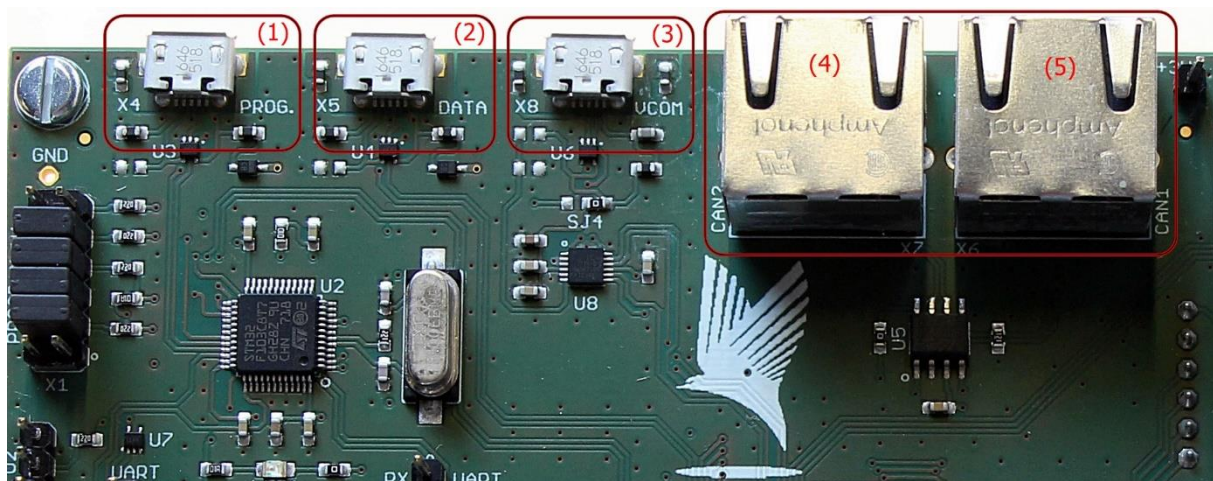


Abbildung 3

APUS Board mit CAN / USB Anschlüssen

Die ersten Schritte

Das Board APUS ist mit einer Testfirmware bestückt und sofort nach Auslieferung betriebsbereit.

Zuerst wird das Board mit Spannung versorgt. Dazu wird das USB Kabel an dem mittleren der drei USB Stecker (X5) mit dem PC oder Laptop verbunden. Anschließend erscheint am Display die Aufforderung zur Kalibrierung. Über USB meldet sich am PC eine Maus als USB-HID-Device an.

Funktionen des Testprogramms

Für die Navigation durch die Testapplikation folgen Sie den Schritten am Display. Zuerst wird der Bildschirm für die Touch-Funktion kalibriert: Für die Kalibrierung und Nutzung des Touch-Displays empfiehlt sich ein Stift, da somit die Genauigkeit erhöht wird. Bewegen Sie zuerst den Stift von unten nach oben (wipe up), anschließend von oben nach unten (wipe down), dann von rechts nach links (wipe left) und schließlich von links nach rechts (wipe right). Führen Sie den Stift immer bis zum Rand des Displays (schwarze Linie). In weiterer Folge steht der Touch für die Eingabe zur Verfügung.



Abbildung 4 Kalibrierung des Touchpanels

Sie gelangen nach erfolgreicher Kalibrierung automatisch in das Hauptprogramm (siehe Abbildung 5).

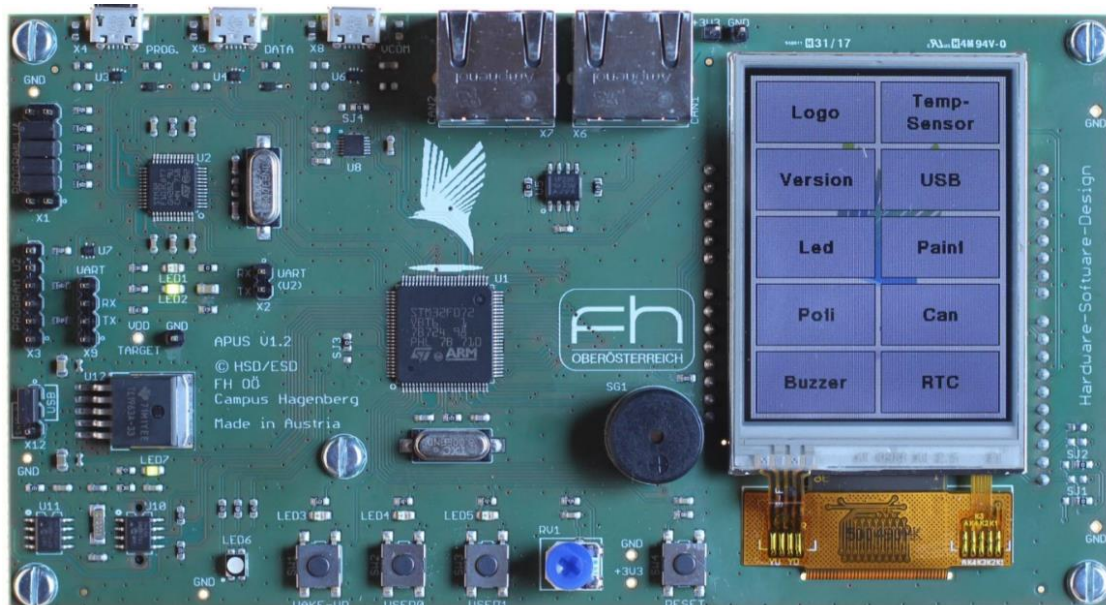


Abbildung 5 Menü Demoapplikation

Anschließend können Sie eines der zehn Testprogramme auswählen.

Bildschirm

Wir starten mit dem ersten Test – der Anzeige von Bitmaps auf dem Display. Dazu drücken Sie links oben auf die Schaltfläche „Logo“. Navigieren Sie mit den Pfeilen, um in das nächste beziehungsweise vorherige Testprogramm zu wechseln. Mit dem rot unterlegten Button „x“ rechts oben, wechseln Sie wieder in das Hauptmenü.



Abbildung 6 Menü Logo und Version

Leuchtdioden

Die Leuchtdiode LED3 (rot), LED4 (orange) und LED5 (grün) können über die Buttons am Bildschirm ein- bzw. ausgeschaltet werden. Über die RGB-Leuchtdiode LED6 können die Farben rot (R), grün (G), blau (B), gelb (RG), cyan (GB), magenta (RB) und weiß (RGB) ausgegeben werden. Der Button „Fading“ aktiviert einen Farbwechsel zwischen den drei Grundfarben.

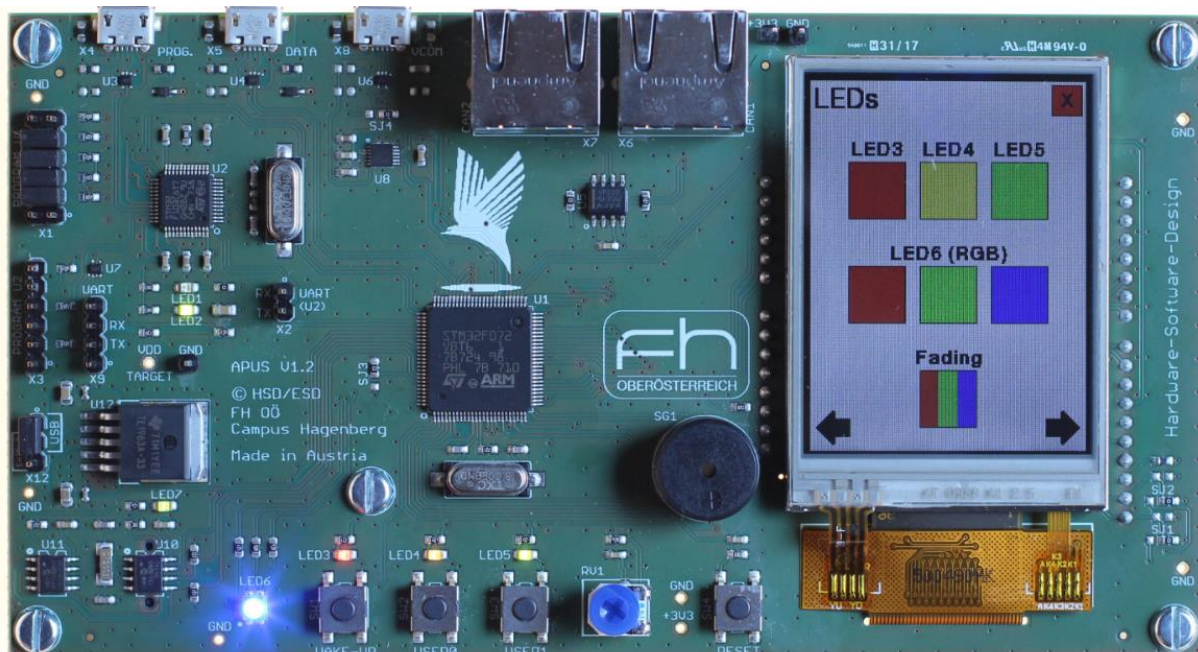


Abbildung 7 Menü LEDs

Potentiometer

Die Einstellungen des Potentiometers RV1 (in der Abbildung rot eingrahmt) können am Display abgelesen werden. Das Potentiometer kann entweder per Hand oder mit einem kleinen Schraubendreher im Wertbereich 0 bis 3,3 Volt verändert werden.



Abbildung 8 Menü Potentiometer mit verschiedenen Werten

Buzzer

In diesem Menü wird der Buzzer aktiviert. Durch Drücken des Buttons „Beep“ wird der Buzzer (rot eingrahmt) für 100ms aktiviert.

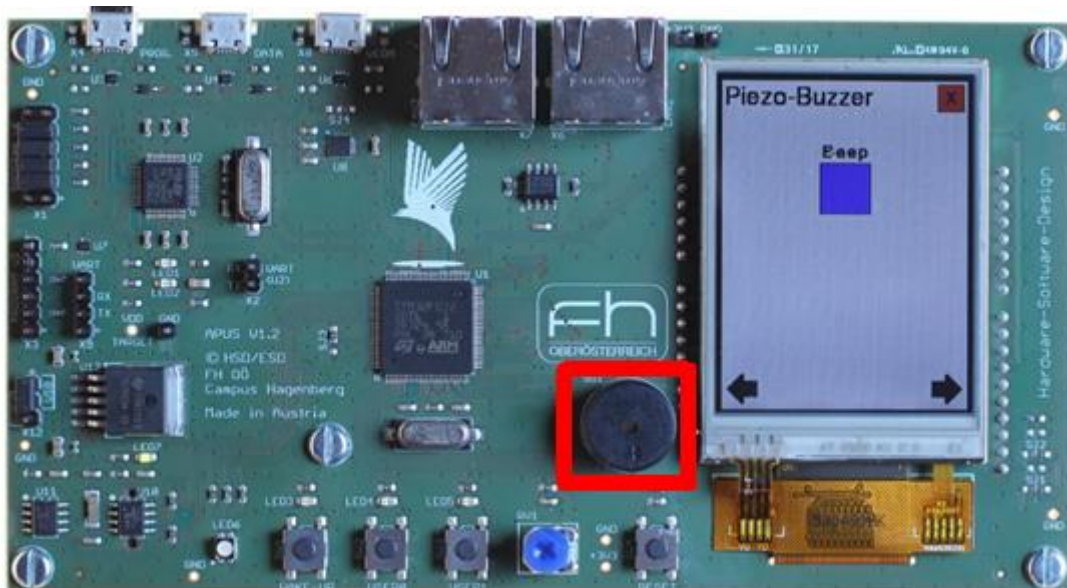


Abbildung 9 Menü Piezo-Buzzer

Temperatursensor

Mit diesem Testprogramm wird der Temperatursensor (rot eingrahmt in Abbildung 10) ausgelesen und der aktuelle Temperaturwert am Display angezeigt. Durch Erhitzen bzw. Kühlen des Sensors kann die Funktion überprüft werden.

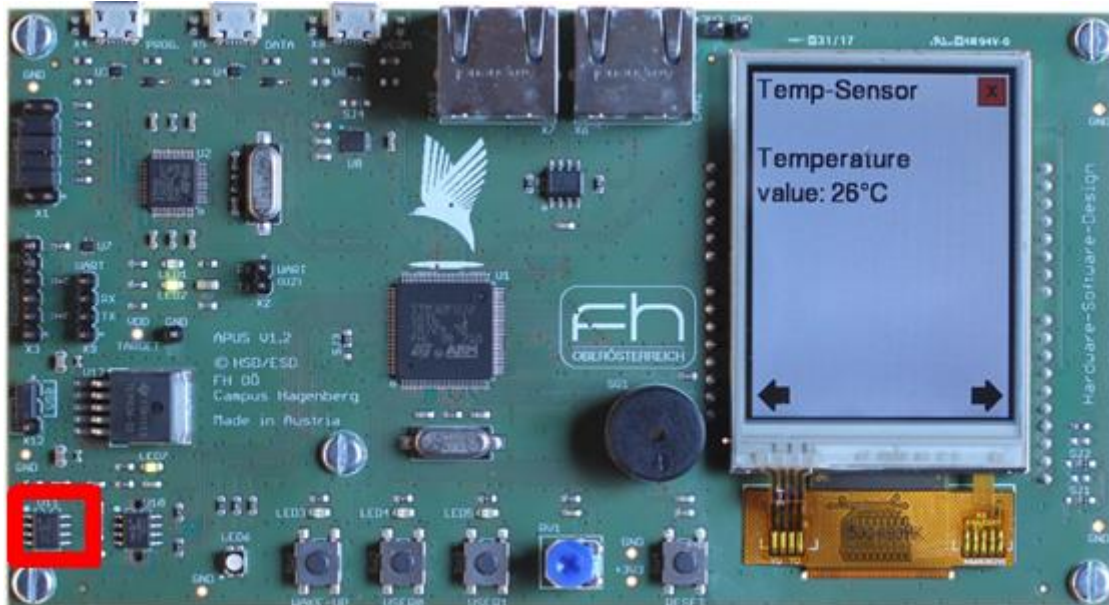


Abbildung 10 Menü Temp-Sensor

USB Mausemulator

Zeichnen Sie mit einem Stift im Feld touchpad und bewegen Sie so den Mauszeiger am PC. Die Felder unten links und rechts entsprechen der linken und rechten Maustaste. Das Feld „Lock“ bedeutet, dass die Maustasten gedrückt bleiben.

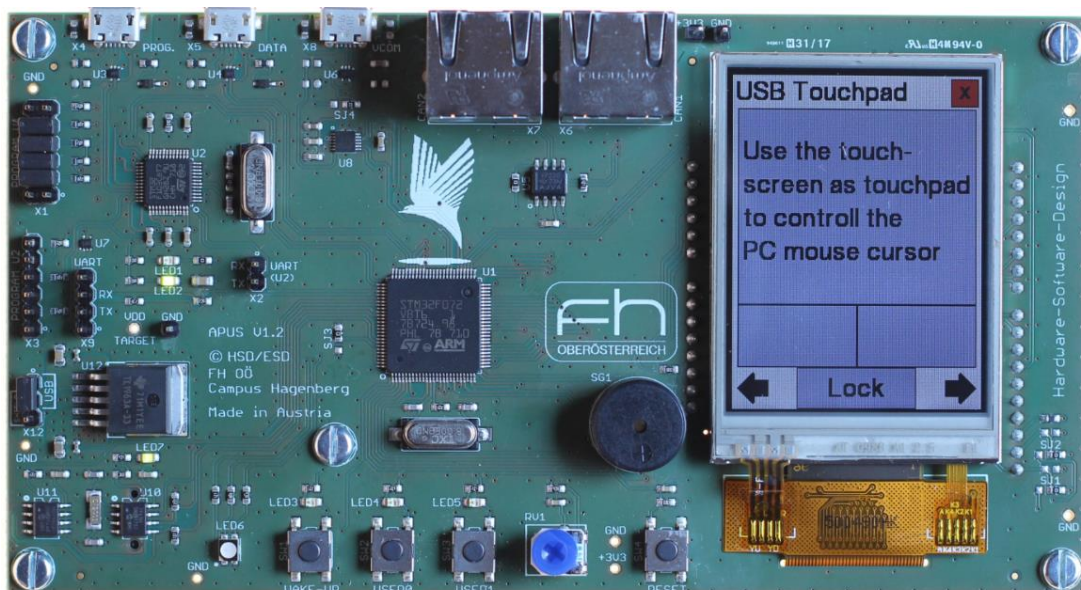


Abbildung 11 Menü USB

Zeichenprogramm

In dieser Demo können Sie mit unterschiedlichen Farben zeichnen. Wählen sie dazu die Farbe aus und zeichnen Sie mit dem Stift Ihre Grafik. Mit dem Button „Clr“ wird der gesamte Zeicheninhalt gelöscht.

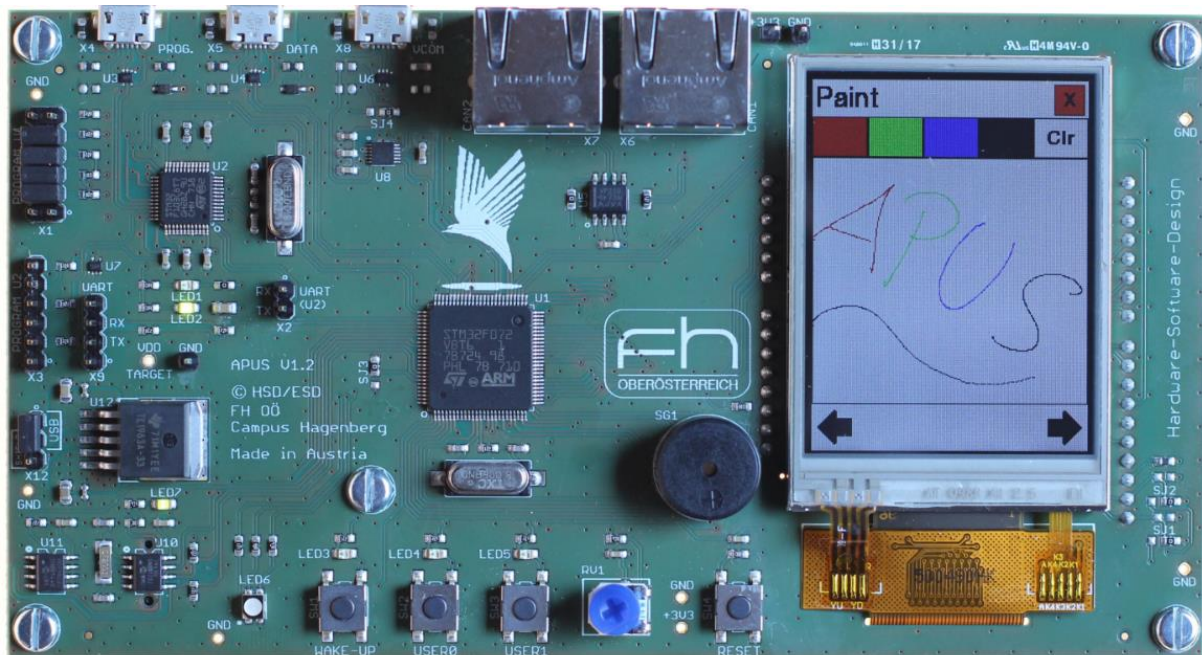


Abbildung 12 Menü Paint

CAN Test

Beim CAN-Test werden zwei APUS-Entwicklungsboards benötigt. Über die Buchsen (1) oder (2) müssen die Boards mit einem Netzkabel mit RJ45 Anschluss verbunden werden. Anschließend werden mit den Buttons am Display die Leuchtdioden LED3, LED4, LED5 des verbundenen Boards gesteuert. Für jede Änderung am anderen Board muss der Button „Send“ gedrückt werden. Ein Counter zählt mit, wie viele Nachrichten gesendet wurden.

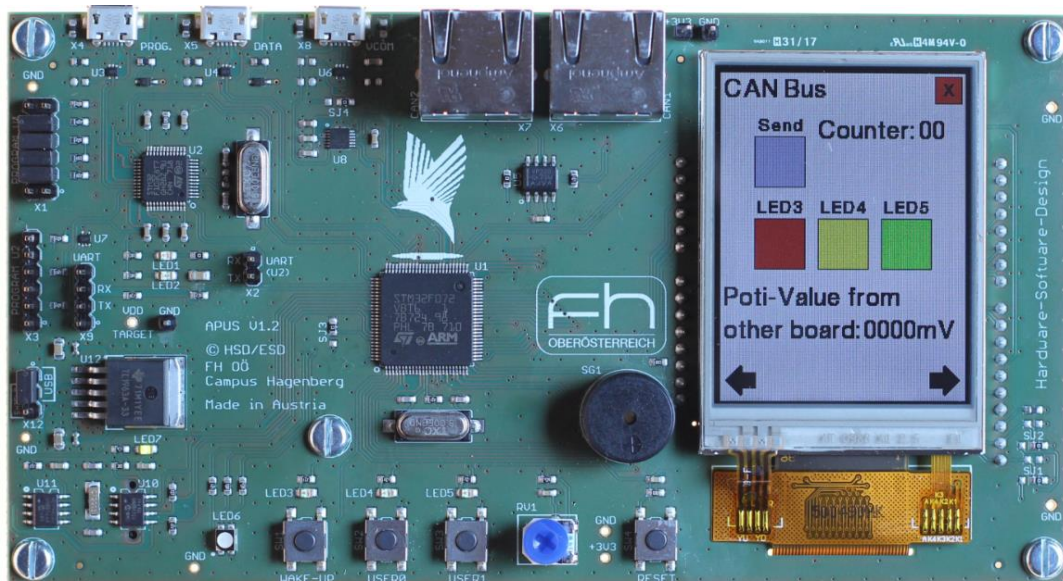


Abbildung 13 Menü CAN

Real Time Clock (RTC)

Die letzte Demo liest die Daten der Echtzeituhr ein und stellt diese am Display dar. Die Startwerte werden über den Touch-Bildschirm eingegeben. In der ersten Zeile werden Stunden, Minuten und Sekunden, in der zweiten Zeile Tag, Monat und Jahr eingegeben. Nach dem Drücken des Start-Buttons beginnt die Echtzeituhr zu laufen. Wenn im Batteriesockel X11 eine CR2032-Batterie platziert wurde, dann wird die RTC über diese Batterie versorgt.

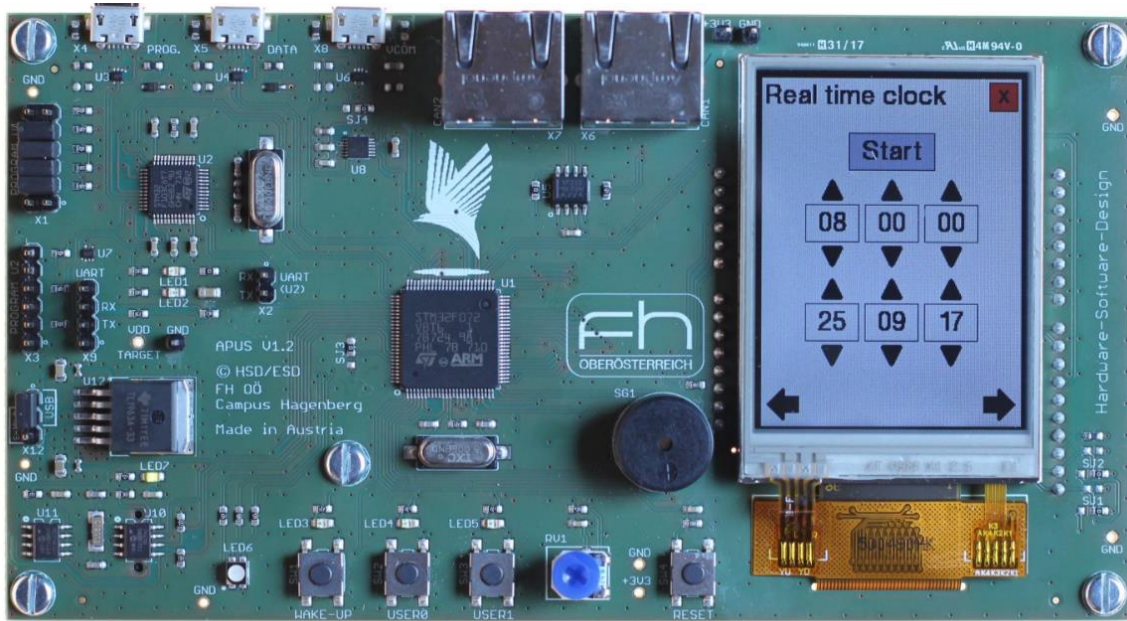


Abbildung 14 Menü RTC

Aufbau APUS

In Abbildung 15 sind die Komponenten von APUS dargestellt, die in diesem Kapitel genauer erklärt werden. Im Mittelpunkt steht der Mikrocontroller STM32F072 mit dem ARM CORTEX M0 Prozessorkern, der mit den Bausteinen und Schnittstellen in Abbildung 15 verbunden ist.

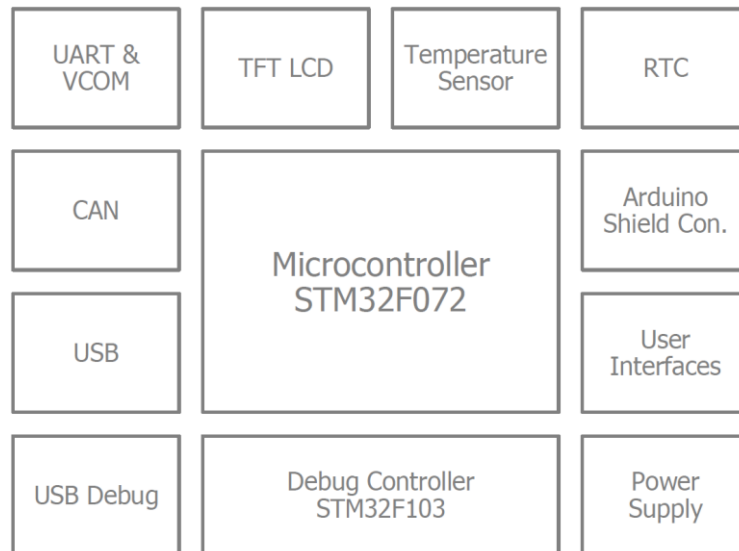


Abbildung 15 Blockschaltbild APUS

Spannungsversorgung

Die Spannungsversorgung von APUS erfolgt über einen der drei USB Ports oder über das Arduino Shield. Die Auswahl erfolgt über den Jumper X18. In der Stellung 1-2 wird die 5 V-Spannungsversorgung von einem der drei USB Ports verwendet, in der Stellung 2-3 wird die Versorgung vom Arduino Shield verwendet. Der maximale Eingangsspannungsbereich des LDOs U12 (Low-Dropout Regulator) beträgt 3,6 -20 V. Die Ausgangsspannung des LDOs beträgt 3,3 V und wird mit zwei Kondensatoren (C52, C53) geglättet.

Mittels Schottky-Dioden (D2 und D3, siehe Abbildung 16) sind die USB-Ports vor Überspannungen vom Arduino Shield Steckplatz und den anderen USB-Ports abgesichert. Die APUS-Elektronik selbst ist nicht gegen Überspannungen vom Arduino Shield geschützt.

Sobald APUS mit Spannung versorgt wird, leuchtet die grüne Leuchtdiode LED7.

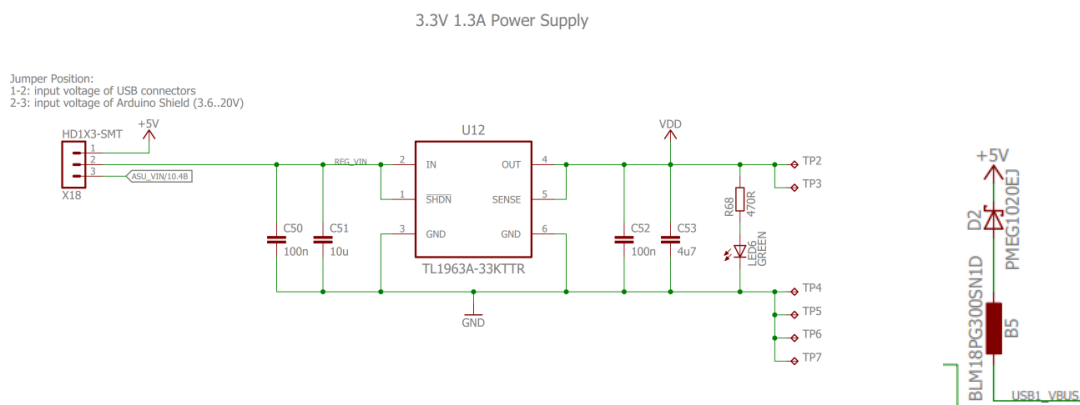


Abbildung 16 Spannungsversorgung

Mikrocontroller STM32F072

APUS verwendet den Mikrocontroller STM32F072VBT6 als Hauptprozessor. Dieser beinhaltet einen ARM CORTEX-M0 Prozessorkern mit integriertem 128 kB Flash und 16 kB RAM Speicher. Ein externer Quarz mit 8 MHz (siehe Abbildung 18) taktet den Prozessor, intern kann die Taktrate bis zu 48 MHz erhöht werden.

Der Prozessor ist mit 87 I/O-Leitungen ausgestattet, die für die Ansteuerung der Peripherie (LED, Taster, Display, Touch, I²C, CAN, UART, USB, Potentiometer, ADC) verwendet werden. In Abbildung 17 ist der Prozessor mit seinen Verbindungen nach außen aufgrund der übersichtlicheren Darstellung in zwei Schaltbildern (U1A und U1B) dargestellt.

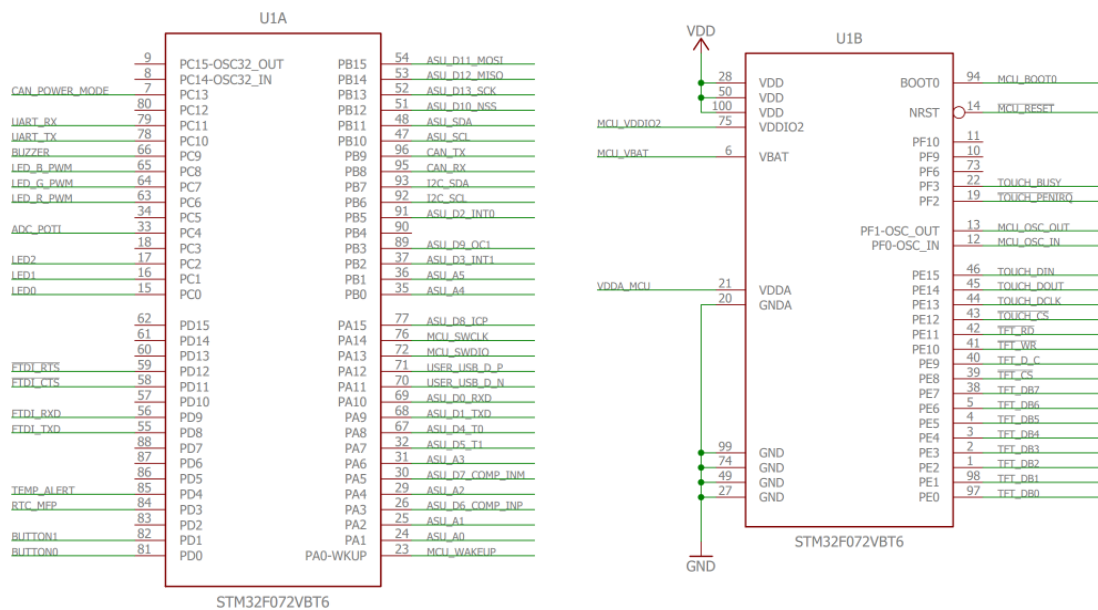


Abbildung 17 Hauptprozessor APUS

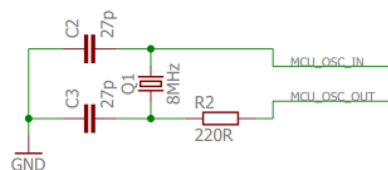


Abbildung 18 Quarz für Hauptprozessor

User Interface

APUS hat neben dem Touch-Display noch einfache Ein- und Ausgabemöglichkeiten, die hier dargestellt sind.

Die drei Taster (User0, User1, Wake-Up) sind durch ein Tiefpassfilter entprellt und direkt mit einem Pin des Mikroprozessors verbunden. Sind die Taster gedrückt, so liegt VDD (3,3V) am Mikroprozessor Pin an, sonst sind diese über einen 10 k-Pull-Down-Widerstand auf Masse gelegt.

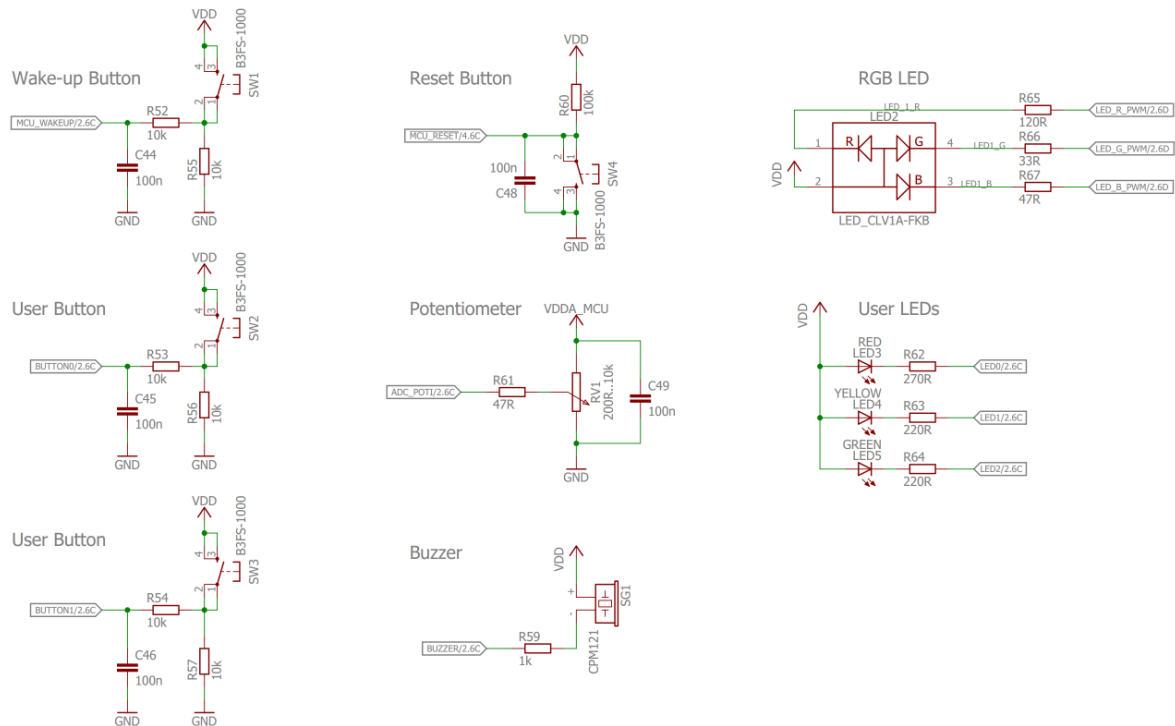


Abbildung 19 User-Interface APUS

Mit dem Reset-Button wird der Mikroprozessor zurückgesetzt und das Programm startet erneut.

Das Potentiometer kann Werte zwischen 0 und 3,3 V (VDDA_MCU) einnehmen. Die analoge Spannung wird vom ADC-Pin des Mikroprozessors eingelesen.

Für die Ausgabe stehen vier optische und ein akustischer Aktor zur Verfügung. Der Buzzer wird über einen Pin des Mikroprozessors ein- bzw. ausgeschaltet. Die Leuchtdioden LED3, LED4, LED5 werden ebenfalls über einen Pin des Mikroprozessors gesteuert. Die RGB-Leuchtdiode LED6 wird über drei Leitungen des Mikroprozessors (R,G,B) angesteuert. Über Pulsweitenmodulation kann die Helligkeit der Leuchtdioden konfiguriert werden.

Temperatursensor und RTC Real Time Clock

Der Temperatursensor MCP9803 sowie die Echtzeituhr MCP79410 sind über den I²C-Bus verbunden. Die Adresse des Temperatursensor ist durch das Verbinden der Pin A0,A1,A2 auf GND mit 0b1001 000 festgelegt.

Der Baustein MCP79410 verfügt neben der Echtzeituhr auch über einen SRAM und EEPROM Bereich. Das EEPROM des Bausteins wird mit 0x1010111 adressiert, RTC bzw. SRAM über die Adresse 0b1101 111 angesteuert.

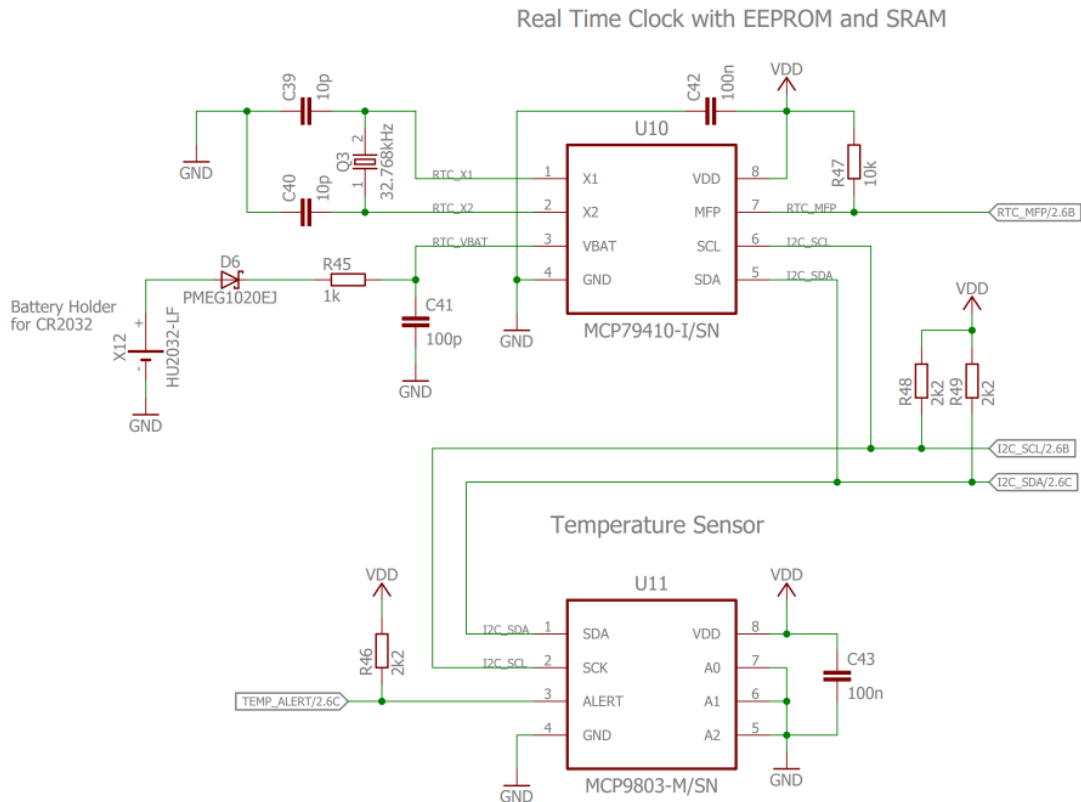


Abbildung 20 User-Interface APUS

Jumper Konfigurationen und Test-Pins

Auswahl der Spannungsversorgung

Mit dem Jumper X12 wird die 5V-Spannungsversorgung eingestellt:

- Stellung 1-2: Versorgung über USB



Abbildung 21 Jumper-Stellung für die Versorgung über USB

- Stellung 2-3: Versorgung über Arduino Shield Steckplatz

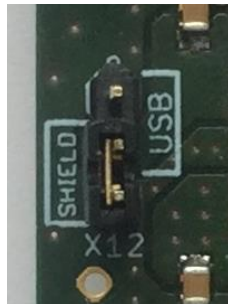


Abbildung 22 Jumper-Stellung für die Versorgung über das Arduino Shield

Konfiguration der Programmierschnittstelle

Mit der Stiftleiste X1 wird festgelegt, ob der Prozessor am Board APUS oder ein externer Prozessor programmiert wird. Die Stiftleiste X3 wird zur Programmierung des Onboard-Programmer (STM32F103) verwendet.

- Stellung 3-4 (SWCLK), 5-6 (GND), 7-8 (SWDIO) und 9-10 (NRST): Programmierung von U1 (STM32F072) über Onboard-Programmer

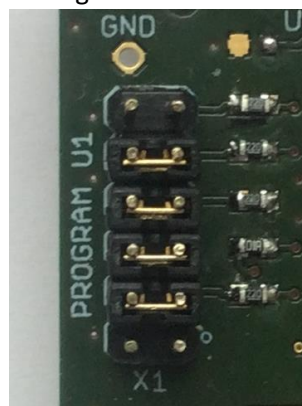


Abbildung 23 Jumper-Stellungen für die Programmierung von U1 über den Onboard-Programmer

- keine Jumper, Anschluss eines Programmierkabels an 4 (SWCLK), 6 (GND), 8 (SWDIO) und 10 (NRST): Programmierung von U1 (STM32F072) über externen Programmer (siehe Abbildung 24 oberes APUS Board)

keine Jumper, Anschluss eines Programmierkabels an 3 (SWCLK), 5 (GND), 7 (SWDIO) und 9 (NRST): Programmierung von externen Controller (siehe Abbildung 24 unteres APUS Board)

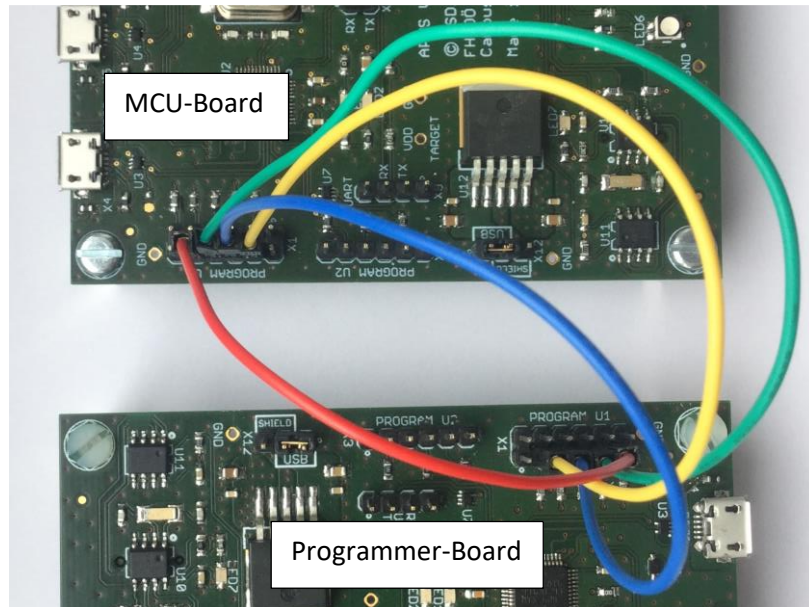


Abbildung 24 Programmierung über den externen Programmer (unten befindet sich das Board mit dem Programmer und oben das Board mit dem zu programmierenden Mikrocontroller)

ST-Link/V2-A Virtual COM

Die ST-Link/V2-A Implementierung auf dem Mikrocontroller U2 (STM32F103) unterstützt einen virtuellen COM Port (VCP). Die Pins TX und RX sind mit der Stiftleiste X2 verbunden. Somit kann über X2 ein USART von U2 mit einem VCP auf dem PC verbunden werden.



Abbildung 25 USART-Schnittstelle für die virtuelle COM Port Unterstützung

Spannungsmesspunkte

Auf dem APUS Board sind mehrere GND- und 3,3 V-Messpunkte verteilt. Abbildung 26 zeigt die Positionen dieser Messpunkte. Die Messpunkte sind jeweils beschriftet. Die 5 V-Spannung kann über die Stiftleiste X12 (siehe *Abbildung 21*) gemessen werden.

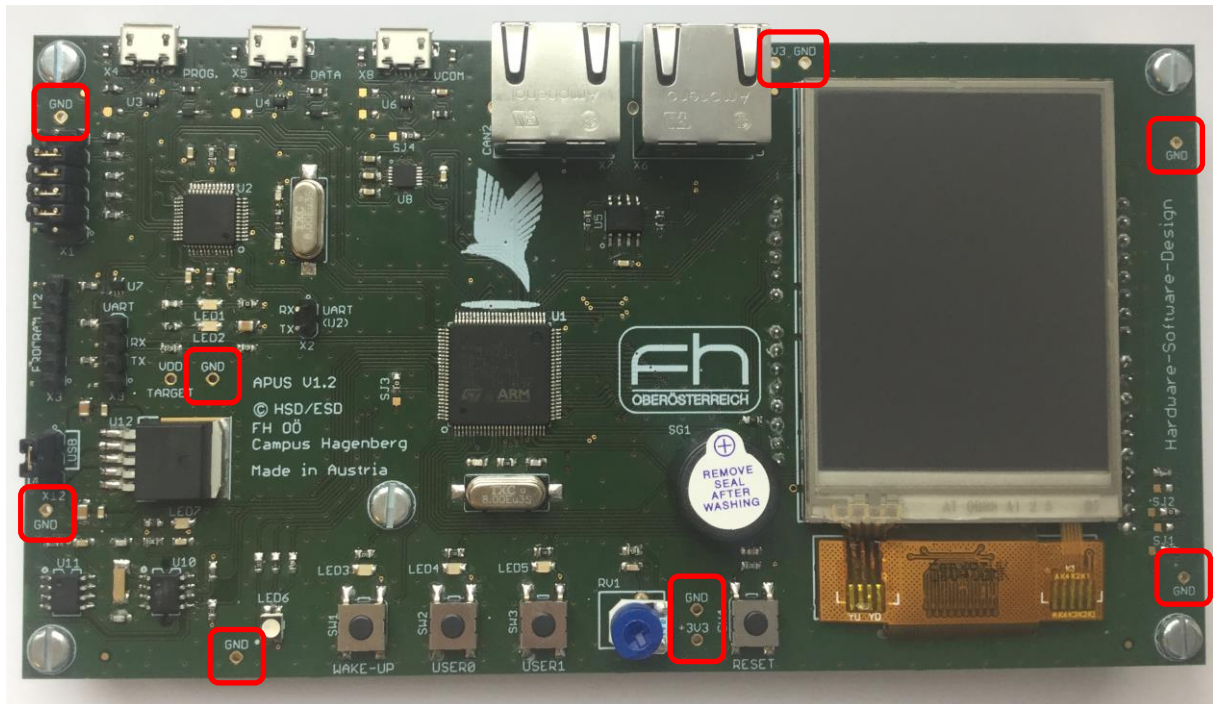


Abbildung 26 Messpunkte

Arduino Shield

APUS ist mit einer Erweiterung für Arduino Shields ausgerüstet. Damit können sehr viele Erweiterungen einfach umgesetzt werden. Untenstehend sind die beiden Arduino Shields WiFi/Bluetooth, IoT Sensor und Ethernet beschrieben.

Arduino WiFi/Bluetooth Shield

Das ISMART Shield kombiniert WiFi und Bluetooth in einem Modul. Für die drahtlose Kommunikation befindet sich auf dem Modul der Broadcom Chip BCM43340 mit integriertem Bluetooth 4.0. Das ISMART Shield ist mit ST Microelectronics CUBE SDK und dem Arduino UNO R3 Steckerlayout kompatibel. Das Inventek eS-WiFi-Modul benötigt kein Betriebssystem und enthält einen integrierten TCP/IP Stack. Das Modul kann mit Inventeks AT-Befehlen (IWIN – Inventek Wireless Interoperability Network) oder mit dem WICED™ (Wireless Internet Connectivity for Embedded Devices) WiFi-Software Entwicklungskit von Broadcom verwendet werden.

Technische Merkmale

- Broadcom Single-Chip, Dual-Band 2,4/5,0 GHz, IEEE 802.11 a/b/g/n mit integriertem Bluetooth 4.0 (BLE-Unterstützung)
- ST Cortex M3 Mikrocontroller (STM32F204)
- U.FL Anschluss für externe Antennen
- Konfigurierbar mit Inventek AT-Befehlen (IWIN)
- Hostschnittstelle: UART, SPI
- Eingangsspannung: 5,0 V
- SPI-Flash für OTA-Updates



Abbildung 27 WiFi/Bluetooth Shield

Arduino Sensor Shield

Das SensorTile Entwicklungskit STEVAL-STLKT01V von STMicroelectronics ist für die Entwicklung und Auswertung einer Vielzahl von Applikationen mit der SensorTile-Platine ausgelegt. Dieses Kit enthält alles was benötigt wird, um Bewegungs-, Umwelt- und akustische Parameter aus der Ferne zu erfassen und zu messen. Die Platine ist so konzipiert, dass es das Prototyping neuer Projektphasen unterstützt und die Startphasen drastisch beschleunigt.

Die SensorTile, das auf das Development Kit gesteckt wird, ist ein kleines (13,5x13,5 mm), quadratisch geformtes IoT-Modul mit leistungsfähigen Verarbeitungsfunktionen. Das Modul verfügt über einen Cortex-M4 80 MHz Mikrocontroller (STM32L476) mit niedrigem Stromverbrauch und Bluetooth Low Energy Konnektivität basierend auf dem BlueNRG Netzwerkprozessor. Der SensorTile verfügt auch über ein breites Spektrum an Bewegungs- und Umwelt-MEMS-Sensoren, darunter auch ein Mikrofon.

Das STEVAL-STLKT01V1 Entwicklungs-Kit enthält eine Reihe von Cradle-Boards, die Hardware-Skalierbarkeit und Software-/Firmware-Bibliotheken und Tools ermöglichen. Mit der gebrauchsfertigen App, ist SensorTile ein echtes IoT-Designlabor.

Technische Merkmale

- STLX01V1 SensorTile-Trägerplatinen-Erweiterung mit USB, Audio-DAC und Arduino-Anschlüssen
- Kompakte STLCR01V1 SensorTile-Trägerplatine mit USB, Ladegerät, Feuchtigkeits- und Temperatursensor, SD-Kartenslot
- SWD-Programmierskabel kompatibel mit ST-Link-Steckverbinder auf STM32 Nucleo-Boards
- Kunststoffgehäuse für SensorTile, Trägerplatine und Akku
- STSW-STLKT01 SensorTile-Firmware unterstützt das Streaming von Sensor-Rohdaten über USB, Datenlogging auf SD-Karte
- BLUEMICROSYSTEM2 STM32Cube-Erweiterungssoftware unterstützt auf die boardinternen Sensoren zugeschnittene Algorithmen
- FP-SNS-ALLMEMS1 STM32 ODE-Funktionspaket, ST BlueMS, BlueST-SDK
- Kompatibel mit STM32 Ökosystem durch STM32Cube-Unterstützung



Abbildung 28 STLX01V1 Sensor Shield

Arduino Ethernet Shield

Mit dem W5500 Ethernet Shield v1.0 wird das APUS Board internetfähig. Die Internetkonnektivität für die Anwendung wird mit einem eigenen Chip hergestellt, der TCP/IP Stack, 10/100 Ethernet MAC und PHY in sich vereint. Das Shield verfügt über zwei Grove-Stecker (I2C und UART) und einen Slot für MicroSD-Karten, sodass Projekte unterstützt werden können, die große Datenmengen vom Grove-Sensor speichern müssen. Das W5500 Ethernet Shield ist so dimensioniert, um ggf. weitere Shields auf diesen zu „stacken“.

Technische Merkmale

- folgende TCP/IP-Protokolle werden unterstützt: TCP, UDP, ICMP, IPv4, ARP, IGMP, PPPoE
- unterstützt 8 unabhängige Slots/Buchsen gleichzeitig
- unterstützt Power-Down-Modus
- unterstützt „Wake on LAN over UDP“
- unterstützt serielle Hochgeschwindigkeits-SPI (SPI MODE 0,3)
- interner Speicher, 32 kBytes, für TX/RX-Puffer
- 10BaseT/100BaseTX Ethernet PHY integriert
- unterstützt Auto Negotiation (Voll- und Halb-Duplex, basiert auf 10 und 100)
- IP-Fragmentierung wird nicht unterstützt
- Betrieb mit 3,3 V; 5 V I/O-Signal-Toleranz
- LEDs für Ausgänge (Voll- /Halb-Duplex, Link, Speed, Active)
- Kartenfach für MicroSD-Karten
- Grove-Anschluss für I²C und UART



Abbildung 29 ETHERNET SHIELD W5500