

MEXLE2020 Modul : MMC_1x1_328PB

Student Group

| First Name | Surname | Matrikel Nr. |
|------------|---------|--------------|
| | | |
| | | |
| | | |

Table of Contents

| | |
|--|---|
| MEXLE2020 Modul : MMC_1x1_328PB | 2 |
| Hardwareübersicht | 2 |
| <i>Human-Machine-Interface</i> | 2 |
| <i>Inter-Platinen Interfaces</i> | 2 |
| Belegung Buchse K1 (links) | 3 |
| Belegung Buchse K2 (rechts) | 3 |
| <i>Eagle-Dateien</i> | 3 |
| Softwareübersicht | 4 |
| Projektübersicht | 4 |
| <i>Aufgabenstellung/Problemstellung</i> | 4 |
| <i>Lösungsansatz</i> | 4 |
| <i>Erstellung der Hardware</i> | 4 |
| <i>Überprüfung/Test</i> | 4 |
| <i>Resume</i> | 4 |

MEXLE2020 Modul : MMC_1x1_328PB

| | |
|-------------------------|---|
| Modultyp | MMC - MEXLE Microcontroller CPU) |
| Versionierung | 1.0 |
| Ersteller | Gerhard Gruhler / Tim Fischer |
| letzte Änderung | 2018-01-03 |
| Kurzbeschreibung | Microcontroller Platine mit ATmega328PB Chip, Quarz, LED und 2 Taster |

Hardwareübersicht



Fig. 1: fertige MMC 1x1 328PB Platine

Die Microcontrollerplatine ([figure 1](#)) nutzt den Chip [ATmega328PB](#). Damit ist es u.a. möglich 2 I2C-, 2 SPI und 8 Analog-Digital-Converter genutzt werden. Details zur Pin-Konfiguration des Chips sind dem verlinkten Datenblatt zu entnehmen, die Pinbelegung des Boards ist unten beschrieben.

Die 1x1 Platine kann als Grundlage für weitere Projekte dienen. Dazu ist sie mit verschiedenen Schnittstellen ausgestattet, die im Folgenden beschrieben werden.

Human-Machine-Interface

 Fig. 2: Human-Machine-Interface der MMC 1x1 328PB Platine

Zur Eingabe sind 2 Taster verbaut (vgl. [figure 2](#)). Diese ermöglichen das Auslösen eines Reset und das Schalten zu Masse. Letzteres bietet bei aktivierten, internen Pull-up Widerstand eine digitale Eingabe.

Weiterhin ist eine LED auf der Platine vorhanden. Diese lässt sich über den Pin 24 aktivieren.

Inter-Platinen Interfaces

 Fig. 3: Board-Board-Interface der MMC 1x1 328PB Platine

Für die Verknüpfung zwischen mehreren Platinen gibt es verschiedene Schnittstellen (siehe [figure 3](#)).

Mit der **Micromatch-Schnittstelle (X1, oben)** lässt sich das Progi verknüpfen. Dieses kann über [ISP](#) per SPI den Chip programmieren.

Die **Pinheader an den Ecken** sind bei diesem Board aktuell nicht elektrisch genutzt. Bei anderen Boards sind diese mit PGND (Power Ground, Masse für Leistungskomponenten) und PVCC (Power Voltage, Spannung für Leistungskomponenten) belegt.

Die **Buchsen links (K1) und rechts (K2)** ermöglichen einen Zugriff auf (fast) alle Pins des Controllers. Bis auf die Pins 7 (XTAL1) und 8 (XTAL2) sind alle Pins verfügbar. Die untersten Pins der Buchse K1 können optional über die Jumper SJ2 und SJ3 auf der Rückseite der Platine entweder auf V+ und GND oder auf Pin 3 und 6 gelegt werden. Ersteres ist für die Kompatibilität der verschiedenen Controllerplatinen notwendig. Letzteres bietet die Möglichkeit die letzten beiden Pins - und damit die I2C-Schnittstelle - anzusprechen. Im Bild ist die Anordnung der Pins auf die Buchsen zu sehen; diese sind für alle Controllerplatinen gleich. Die beiden Buchsen ermöglichen Hook-up-Platinen, welche auf den Controllerplatinen aufbauen. Eine Spannungsversorgung der Controllerplatine ist auch über die Hook-ups möglich.

Die **Stecker unten (JP1)** verknüpfen die Platine mit der Basisplatine. Damit ist die Spannungsversorgung über die Basisplatine möglich. Die Pins V+ und GND dieses Steckers wird für die Versorgung des Controllers genutzt. Die Spannung V- wird auf dem Board nicht direkt genutzt, aber an Hook-ups (über Buchse K2) weitergeleitet. Über diesen Stecker ist auch eine I2C-Verbindung zwischen verschiedene Platinen über die Basisplatine möglich. Weiterhin gibt es einen Reset-Pin, welcher einen zentralen Reset aller Platinen auf der Basisplatine erlaubt.

Belegung Buchse K1 (links)

Belegung Buchse K2 (rechts)

| Functions | Pin | Pin | Functions | Functions | Pin | Pin | Functions |
|-------------------------------|-------|-----|--------------------------------|--------------------|-------|-----|-------------------------|
| PTC X0 Y8, OC3A, RXD0 | Pin30 | 31 | PTC X1 Y9, OC4A, TXD0 | VCC | Pin4 | 21 | AGND |
| PTC X2 Y10, INT0, OC3B / OC4B | Pin32 | 1 | PTC X2 Y11, INT1, OC2B | GND | Pin5 | 20 | AREF |
| PTC X4 Y12, T0, XCK0 | Pin2 | 9 | PTC X5 Y13, OC0B,T1 | - | NC | 18 | AVCC |
| PTC X6 Y14, AIN0, OC0A | Pin10 | 11 | PTC X7 Y15, AIN1 | - | NC | 29 | RESET |
| PTC X10 Y18, CLK0, ICP1 | Pin12 | 13 | PTC X11 Y19, OC1A | PTC Y5, ADC5, SCL0 | Pin28 | 27 | PTC Y4, ADC4, SCL1 |
| PTC X12 Y20, OC1B, SS0 | Pin14 | 15 | PTC X13 Y21, OC2A, TXD1, MOSI0 | PTC Y3, ADC3 | Pin26 | 25 | PTC Y2, ADC2 |
| PTC X14 Y22, RXD1, MISO0 | Pin16 | 17 | PTC X15 Y23, XCK1, SCK0 | PTC Y1, ADC1, SCK1 | Pin24 | 23 | PTC Y0, ADC0, MISO1 |
| PTC X8 Y16, ACO, ICP4, SDA1 | | | | PTC Y7, T3, MOSI1 | Pin22 | 19 | PTC Y6, ADC6, ICP3, SS1 |

Eagle-Dateien

Die aktuellen Eagledateien und Vorversionen sind [hier in Redmine](#) zu finden.

Als Ausgangspunkt können folgende Dateien genutzt werden:

- [mexle328pb_1.2.brd](#)
- [mexle328pb_1.2.sch](#)

Softwareübersicht

Die Software ist abhängig von dem gewünschten Projekt zu verwenden.

Projektübersicht

Aufgabenstellung/Problemstellung

Die Aufgabenstellung war eine kompakte Controllerplatine zu erstellen, welche als Basis für unterschiedlichste Projekte genutzt werden kann. Dabei sollte Wert auf Kompatibilität und größtmögliche Ausnutzung des verwendeten Microcontrollers werden.

Lösungsansatz

1. Wie wurde das Problem angegangen?
2. Auswahl der Bauteile

Erstellung der Hardware

1. Schaltplan
2. Board
3. auf was wurde Wert gelegt?

Welche Hürden wurden genommen?

Überprüfung/Test

1. [Inbetriebnahme-Dokumentation](#)
2. [Aufbau vom Board](#)

Resumee

1. Fazit
2. Verbesserungsvorschläge (Ausblick)

From:

<https://mexle.te.hs-heilbronn.de/> - **MEXLE Wiki**

Permanent link:

https://mexle.te.hs-heilbronn.de/mexle2020/mmc_1x1_328pb?rev=1598641009



Last update: **2021/05/09 10:02**