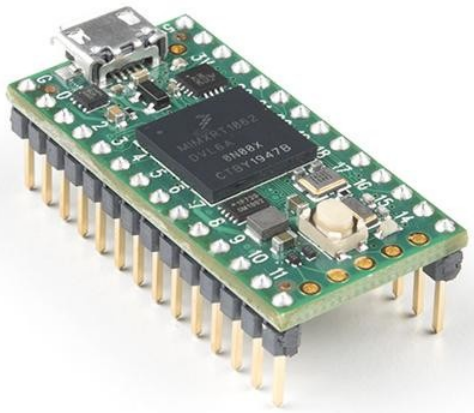
Teensy 4.0, mit Headern



PJRC

Artikel-Nr.:	DEV-16997
Hersteller:	PJRC
Herkunftsland:	USA
Zolltarifnummer:	85423111
Gewicht:	0.01 kg

Teensy 4.0 verfügt über einen ARM Cortex-M7-Prozessor mit 600MHz und einen NXP iMXRT1062-Chip, den schnellsten heute verfügbaren Mikrocontroller. Teensy 4.0 hat die gleiche Größe und Form wie Teensy 3.2 und behält die Kompatibilität mit den meisten Pin-Funktionen von Teensy 3.2 bei. Das Beste an dieser Version von Teensy 4.0 ist, dass sie bereits mit Headern versehen ist. Es ist kein Löten erforderlich, so dass Sie so schnell wie möglich loslegen können!

Beim Betrieb mit 600 MHz verbraucht Teensy 4.0 etwa 100 mA Strom. Teensy 4.0 bietet Unterstützung für dynamische Taktskalierung. Im Gegensatz zu traditionellen Mikrocontrollern, bei denen eine Änderung der Taktrate zu falschen Baudraten und anderen Problemen führt, sind die Hardware von Teensy 4.0 und die Software-Unterstützung für Arduino-Timing-Funktionen so ausgelegt, dass dynamische Geschwindigkeitsänderungen möglich sind. Serielle Baudraten, Audio-Streaming-Sampleraten und Arduino-Funktionen wie delay() und millis() sowie Teensyduino-Erweiterungen wie IntervalTimer und elapsedMillis arbeiten weiterhin korrekt, während die CPU ihre Geschwindigkeit ändert. Teensy 4.0 bietet auch eine Funktion zum Abschalten der Stromversorgung. Durch Anschluss eines Tasters an den On/Off-Pin kann die 3,3V-Stromversorgung durch Halten des Tasters für 5 Sekunden komplett abgeschaltet und durch einen kurzen Tastendruck wieder eingeschaltet werden. Wenn eine Knopfzelle an VBAT angeschlossen wird, behält die RTC des Teensy 4.0 auch bei ausgeschalteter Stromversorgung Datum & Uhrzeit im Auge. Teensy 4.0 kann auch übertaktet werden, weit über 600MHz hinaus!

Der ARM Cortex-M7 bringt viele leistungsstarke CPU-Funktionen in eine echte Echtzeit-Mikrocontroller-Plattform. Der Cortex-M7 ist ein Dual-Issue-Superscaler-Prozessor, d.h. der M7 kann zwei Instruktionen pro Taktzyklus ausführen, und das bei 600MHz! Natürlich hängt die gleichzeitige Ausführung von zwei Befehlen von der Anordnung der Befehle und Register durch den Compiler ab. Erste Benchmarks haben gezeigt, dass von Arduino kompilierter C++ Code dazu neigt, zwei Instruktionen in etwa 40% bis 50% der Zeit auszuführen, während er numerisch intensive Arbeit mit Integern und Zeigern ausführt. Der Cortex-M7 ist der erste ARM-Mikrocontroller, der Verzweigungsvorhersage verwendet. Auf M4 benötigen Schleifen und anderer Code, der viel verzweigt, drei Taktzyklen. Beim M7 entfernt die Verzweigungsvorhersage diesen Overhead, nachdem eine Schleife ein paar Mal ausgeführt wurde, so dass die Verzweigungsanweisung in nur einem Taktzyklus ausgeführt werden kann.

Tightly Coupled Memory ist ein spezielles Feature, das dem Cortex-M7 einen schnellen Single-Cycle-Zugriff auf den Speicher über ein Paar 64 Bit breite Busse ermöglicht. Der ITCM-Bus bietet einen 64-Bit-Pfad zum Abrufen von Befehlen. Der DTCM-Bus ist eigentlich ein Paar von 32-Bit-Pfaden, die es dem M7 ermöglichen, bis zu zwei separate Speicherzugriffe im selben Zyklus durchzuführen. Diese extrem schnellen Busse sind vom AXI-Hauptbus des M7 getrennt, der auf andere Speicher und Peripheriegeräte zugreift. Auf 512K Speicher kann als eng gekoppelter Speicher zugegriffen werden. Teensyduino weist Ihren Arduino-Skizzencode automatisch dem ITCM zu und alle Nicht-Malloc-Speicherverwendung dem schnellen DTCM, es sei denn, Sie fügen zusätzliche Schlüsselwörter hinzu,

Alle hier aufgeführten Namen und Zeichen sind Eigentum des jeweiligen Herstellers. Druckfehler, Änderungen & Irrtümer vorbehalten.

um die optimierte Vorgabe zu überschreiben. Speicher, auf den nicht über die eng gekoppelten Busse zugegriffen wird, ist für den DMA-Zugriff durch Peripheriegeräte optimiert. Da der Großteil des Speicherzugriffs von M7 über die beiden eng gekoppelten Busse erfolgt, haben leistungsfähige DMA-basierte Peripheriegeräte einen hervorragenden Zugriff auf den Nicht-TCM-Speicher für hocheffiziente E/A.

Der Cortex-M7-Prozessor von Teensy 4.0 enthält eine Fließkommaeinheit (FPU), die sowohl 64-Bit-"Double" als auch 32-Bit-"Float" unterstützt. Bei der M4-FPU auf Teensy 3.5 & 3.6, und auch Atmel SAMD51 Chips, ist nur 32 Bit "float" hardwarebeschleunigt. Jegliche Verwendung von Double, Double-Funktionen wie log(), sin(), cos() bedeutet langsame softwareimplementierte Mathematik. Teensy 4.0 führt alle diese Funktionen mit FPU-Hardware aus.

Merkmale:

- ARM Cortex-M7 mit 600MHz
- 1024K RAM (512K sind fest gekoppelt)
- 2048K Flash (64K reserviert für Recovery & EEPROM-Emulation)
- 2 USB-Anschlüsse, beide 480MBit/sec
- 3 CAN Bus (1 mit CAN FD)
- 2 I2S Digital Audio
- 1 S/PDIF-Digital-Audio
- 1 SDIO (4 Bit) nativ SD
- 3 SPI, alle mit 16 Wort FIFO
- 3 I2C, alle mit 4 Byte FIFO
- 7 Serielle, alle mit 4 Byte FIFO
- 32 Allzweck-DMA-Kanäle
- 31 PWM-Pins
- 40 digitale Pins, alle interruptfähig
- 14 analoge Pins, 2 ADCs auf dem Chip
- Kryptographische Beschleunigung
- Zufallszahlengenerator
- RTC für Datum/Uhrzeit
- Programmierbares FlexIO
- Pixelverarbeitungs-Pipeline
- Peripherie-Quertriggerung
- Ein-/Ausschaltmanagement
- Vorgelötete Stiftleisten

Dokumente:

- Handbuch (MIMXRT1062 DVL6A)
- Teensy Schnellstart
- Teensyduino Software
- Teensy Hilfeseite und FAQ

Weitere Bilder:

