Teensy 3.1, ARM Development Board

New Features

Teensy 3.1 has the same size, shape & pinout; is compatible with code written for Teensy 3.0, and is well supported on the Arduino IDE using Teensyduino. Teensy 3.1 adds several new features, due to an upgraded microcontroller. Here are the highlights. Atable of technical specifications is show below.

More Memory For Awesome Projects

The RAM has quadrupled, from 16K to 64K. While 16K is plenty for nearly all Arduino libraries, 64K allows for more advanced applications. Icons and graphics for color displays and audio effects requiring delays, like reverb and chorus, will become possible on Teensy 3.1.
Flash memory has also doubled, to 256K, and provides double the memory bandwidth.

5 Volt Tolerance on Digital Inputs

Today most new chips use 3.3V signals, but many legacy products output 5 volt digital signals. These can now be directly connected to Teensy 3.1's digital inputs. All digital pins are 5 volt tolerant on Teensy 3.1. However, the analog-only pins (A10-A14), AREF, Program and Reset are 3.3V only.

12 Bit Analog Output

Teensy 3.1 has a proper analog output. You can always filter PWM, but true analog output responds rapidly. The output is created by the stable reference voltage, so it's doesn't vary if your power supply voltage changes slightly.

```
// Simple DAC sine wave test on Teensy 3.1

float phase = 0.0;
float twopi = 3.14159 * 2;
elapsedMicros usec = 0;

void setup() {
  analogWriteResolution(12);
}

void loop() {
  float val = sin(phase) * 2000.0 + 2050.0;
  analogWrite(A14, (int)val);
  phase = phase + 0.02;
  if (phase >= twopi) phase = 0;
  while (usec < 500) ; // wait
  usec = usec - 500;
}
```

Simply use analogWrite() on the A14 pin for true analog output.

Two Analog to Digital Converters

With analogRead(), you can measure many signals, but only 1 at a time. Teensy 3.1 has a second ADC, so you can concurrently measure 2 analog signals. For stereo audio or power monitoring, sampling both left and right, or voltage and current at the same moment is very beneficial.
PJRC is developing an advanced analog read function for Arduino, to make using these new analog input capabilities very easy.

CAN Bus

Controller Area Network is a communication used in automobiles. A new library to support Teensy 3.1's CAN controller is planned.

Technical Specifications

Feature	Teensy 3.0	Teensy 3.1	Units
Processor	MK20DX128VLH5	MK20DX256VLH7	
Core	Cortex-M4	Cortex-M4	
Rated Speed	48	72	MHz
Overclockable	96	96	MHz
Flash Memory	128	256	kbytes
Bandwidth	96	192	Mbytes/sec
Cache	32	256	Bytes
RAM	16	64	kbytes
EEPROM	2	2	kbytes
Direct Memory Access	4	16	Channels

Digital I/O	34	34	Pins
Voltage Output	3.3V	3.3V	Volts
Voltage Input	3.3V Only	5V Tolerant	Volts
Analog Input	14	21	Pins
Converters	1	2	
Resolution	16	16	Bits
Usable	13	13	Bits
Prog Gain Amp	0	1	
Touch Sensing	12	12	Pins
Comparators	2	3	
Analog Output	0	1	Pins
DAC Resolution	-	12	Bits
Timers	11 Total	12 Total	
FTM Type	2	3	
PWM Outputs	10	12	Pins
PDB Type	1	1	
CMT (infrared) Type	1	1	
LPTMR Type	1	1	
PIT (interval) Type	4	4	
Systick	1	1	
RTC (date/time) **	1	1	
Communication			
USB	1	1	
Serial	3	3	
With FIFOs	1	2	
High Res Baud	3	3	
Fast Clock	2	2	
SPI	1	1	
With FIFOs	1	1	
I2C	1	2	
CAN Bus	0	1	
I2S Audio	1	1	
FIFO Size	4	8	

** RTC requires a 32.768 kHz crystal & 3V battery. See the Time Library for details.

Pinouts

Teensy 3.1 is meant to be a drop-in replacement for Teensy 3.0. Only 1 pin has different functionality. The reset pin was replaced by A14/DAC, which you can use for true analog output, or as another analog input.

Pins 3 and 4 gained CAN bus functions.

Pins 6, 9, 15, and 20-21 can still have SPI chip select capability, but the labels were removed to simplify the pinout card. Currently no libraries use the native chip selects. However, a new SPIFIFO library is in development.

On the back side, 2 pins gained new PWM functionality and 6 pins have new analog input capability. A second I2C port is also added.

Reset Signal

Teensy 3.0 supported both a Program signal, to reboot into bootloader mode, and a traditional Reset signal, for a conventional reboot that restarts your application. The Reset signal can be found on a new test point on the bottom side of Teensy 3.1.