

ATMEGA4809 Breadboard Kit

CHIP: ATMEGA4809-PF

Parts: 19

Kit Description:

This ATMEGA4809 Breadboard kit allows you to build out your very own MCU on a breadboard/protoboard. The ATMEGA4809 includes 48KB of Flash and 6KB SRAM. The chip runs from 1.8-5.5v and can reach a top frequency of 20MHz (voltage dependent). The ATMEGA4809 is a part of the newer megaAVR 0-series. Programming is available via UPDI, which is a single line serial connection. If you use the USB-UART programmer, by connecting RX to UPDI pin, and TX to UPDI through a 4.7kΩ series resistor, it functions as a USB-UPDI programmer. This kit is ideal for systems with memory requirements surpassing Arduino UNO capacities, space-bound systems and most importantly: education.

Happy building!

ATMEGA4809 PINOUT:

PIN	NAME	DOES THIS	PIN	NAME	DOES THIS
1	PC0	General IO, TX	21	PE2	General IO, AIN10, SCK
2	PC1	General IO, RX	22	PE3	General IO, AIN11, SS
3	PC2	General IO, I2C	23	PF0	TX
4	PC3	General IO, I2C	24	PF1	RX
5	VDD	+3.3v (Can be +5v)	25	PF2	General IO, AIN12, I2C
6	GND	Ground	26	PF3	General IO, AIN13, I2C
7	PC4	General IO, TX, PWM	27	PF4	General IO, AIN14, TX
8	PC5	General IO, RX, PWM	28	PF5	General IO, AIN15, RX
9	PD0	General IO, AIN0, PWM	29	PF6 (Reset)	General IO, RESET
10	PD1	General IO, AIN1, AC0	30	UPDI	UPDI Programming
11	PD2	General IO, AIN2, AC0	31	VDD	+3.3v (Can be +5v)
12	PD3	General IO, AIN3, AC0	32	GND	Ground

PIN	NAME	DOES THIS	PIN	NAME	DOES THIS
13	PD4	General IO, AIN4, AC0	33	PA0	EXTCLK, TX, WO0
14	PD5	General IO, AIN5, AC0	34	PA1	RX, WO1
15	PD6	General IO, AIN6, AC0	35	PA2	General IO, I2C, WO2
16	PD7	General IO, AIN7, VREFA	36	PA3	General IO, I2C, WO3
17	AVDD	+3.3v (Can be +5v)	37	PA4	General IO, MOSI, TX
18	GND	Ground	38	PA5	General IO, I2C, MISO, RX
19	PE0	General IO, MOSI, AIN8	39	PA6	General IO, I2C, SCK
20	PE1	General IO, MISO, AIN9	40	PA7	General IO, SS

This pinout is simplified, for full pinout, see datasheets below

CHIP SPECIFICATIONS:

Core: 8 Bit AVR RISC, 40-pin DIP Package, up to 20MHz (internal oscillator)

Memory: 48KB Flash, 6KB SRAM, 256B EEPROM

Power: 1.8-5.5v, 0.2mA active draw @ 1 MHz 1.8v, 0.1uA at 1.8v power-down mode (Typical values)

IO Peripherals: 33 GPIOs, 11 PWMs, 1 10-bit ADC (multiple channels), 4 USART channels, 1 SPI, 1 I2C/TWI

Other: OSC20M internal oscillator, 16-bit Type A Timer/Counter, 4 Type Bs, 1 RTC

Typical Maximum Clock Frequency:

- 5MHz @ 1.8v
- 10MHz @ 2.7v
- 20MHz @ 4.5v+

Parts in kit:

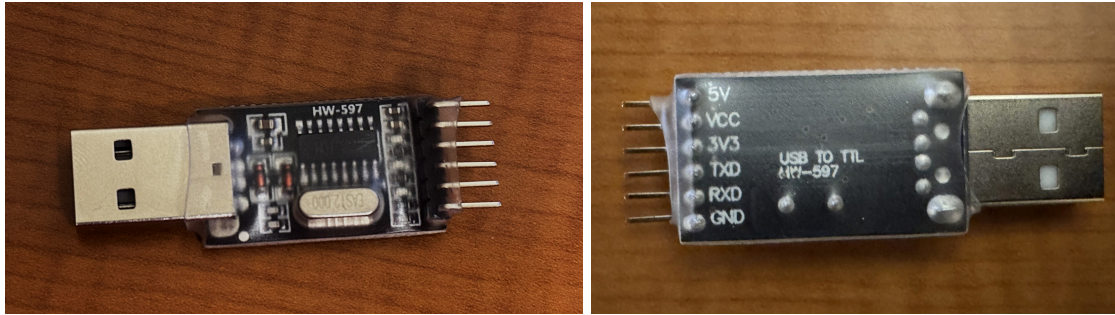
COMPONENT	# Included	JOB
ATMEGA4809	1	Microcontroller
3.3v 0.8A LDO	1	Voltage Regulator (3.3v)
NE555P (Timer)	1	Programmable Timer & Clock

COMPONENT	# Included	JOB
25AA320A (EEPROM)	1	Non Volatile SPI Memory
100nF Ceramic Capacitor	8	Local Decoupling
1uF Ceramic Capacitor	2	Local Decoupling
47uF Electrolytic Capacitor	1	Bulk Decoupling
100uF Electrolytic Capacitor	2	Bulk Decoupling
0.5A Polyfuse	1	VDD Overcurrent Protection
1A Schottky Diode	1	Low Forward Voltage Diode
Ferrite Bead	1	Analog VDD Filtering
33 Ohm Resistor	5	Series resistors for high speeds
470 Ohm Resistor	3	LED resistors
1k Ohm Resistor	3	Pull up/down
4.7k Ohm Resistor	5	Pull up/down
10k Ohm Resistor	5	Pull up/down
Green LED	3	Power good, general use, IO's
Tactile Switch	2	Reset/General Purpose
USB to UART Programmer	1	USB-UART programmer

PROGRAMMING:

The typical method of programming the ATMEGA4809 is over UPDI. Included in this kit is a dedicated USB to UART converter, which breaks out 5v, 3.3v, VCC, GND, RX and TX. First, select your voltage by connecting VCC to the bus you're using (you should also use the USB to power your ATMEGA & circuit if current requirements allow). Next, always ensure you have a common ground. Lastly, connect RX to UPDI and TX to UPDI but through a 4.7k Ω series resistor.

This chip does not NEED a bootloader, so use ArduinoIDE and MegaCoreX, and select ATMEGA4809 via UPDI. Be sure to specify 40 pin standard pinout and select SerialUPDI programming in the Arduino IDE, MegaCoreX wont auto-detect it. Lastly, upload your sketch!



Example Layout:

1. Set up

Place the MCU on the center of the breadboard, giving each pin its own row. Connect VDD to your power rail, and AVDD through a ferrite bead to your power rail. Be sure to also connect all ground pins to your ground rail. On both VDD and AVDD pins, place 100nF capacitors from power to the ground rail. On the VDD rail, place a 47uF electrolytic capacitor, being mindful of polarity, along with a 1uF ceramic capacitor. You can choose which voltage to run your ATMEGA4809 off of, depending on specs you want. The USB to UART programmer gives both 3.3v and 5v options, the kit also comes with a 3.3v LDO. Optionally, you can choose to use a polyfuse in series, and the diode for reverse polarity and over current protections.

2. Programming

Use the USB to UART programmer by connecting RX to UPDI and TX through a 4.7k Ω to UPDI. Use MegaCoreX boards and upload sketches via ArduinoIDE. Be mindful of common grounds, and the fact that your MCU voltage & programming logic levels need to match.

See above for more detailed programming instructions.

3. Application

To use the code featured in the example code section of this datasheet, connect 1 side of a tactile button to GND and the other side to PD2

Example Code (MCU + Button):

```
/*
 * ATMEGA4809 Dev Kit -- Example Sketch: Button Press Counter
 * -----
 * Press the tactile button. Each press prints the press number
 * and the elapsed time (in ms) since the board booted.
 *
 * WIRING:
 * Tactile Button -> PD2, other side to GND (uses internal pull-up)
 */

const uint8_t BUTTON_PIN = PIN_PD2;

int pressCount = 0;
bool lastState = HIGH;

void setup() {
```

```

Serial.begin(9600);
pinMode(BUTTON_PIN, INPUT_PULLUP);
}

void loop() {
  bool currentState = digitalRead(BUTTON_PIN);

  if (lastState == HIGH && currentState == LOW) {
    pressCount++;
    Serial.print("Press #");
    Serial.print(pressCount);
    Serial.print(" - Elapsed time: ");
    Serial.print(millis());
    Serial.println(" ms");
    delay(50); // basic debounce
  }

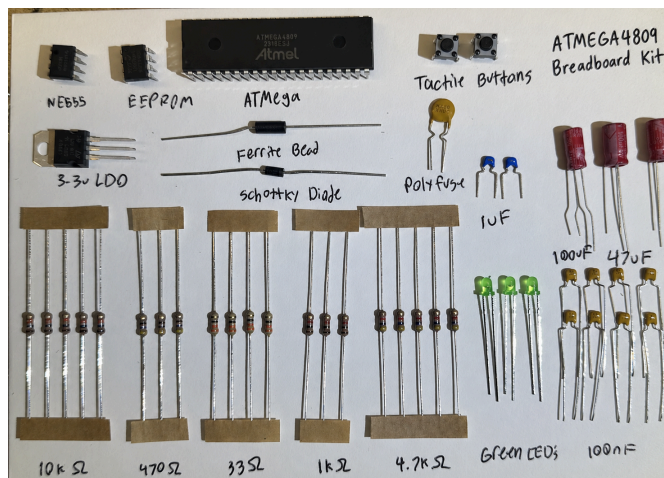
  lastState = currentState;
}

```

Applications:

- Embedded systems prototyping & education
- IoT sensor nodes
- Robotics & motor control
- Lower-power applications
- Custom Arduino-alternative projects
- Space sensitive system

PHOTOS:



Datasheets & Pinout Images:

[\[ATMEGA4809 Datasheet\]](#)

[\[NE555 Datasheet\]](#)

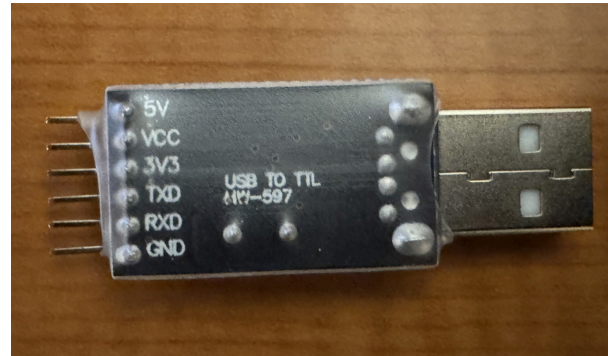
[\[25AA320A Datasheet\]](#)

[\[RXEF050 Datasheet\]](#)

[\[1N5819 Datasheet\]](#)

[\[LD1117V33C Datasheet\]](#)

PC0	1	40	PA7
PC1	2	39	PA6
PC2	3	38	PA5
PC3	4	37	PA4
VDD	5	36	PA3
GND	6	35	PA2
PC4	7	34	PA1
PC5	8	33	PA0 (EXTCLK)
PD0	9	32	GND
PD1	10	31	VDD
PD2	11	30	UPDI
PD3	12	29	PF6 (RESET)
PD4	13	28	PF5
PD5	14	27	PF4
PD6	15	26	PF3
PD7	16	25	PF2
AVDD	17	24	PF1 (TOSC2)
GND	18	23	PF0 (TOSC1)
PE0	19	22	PE3
PE1	20	21	PE2



On the left: Official ATMEGA4809 Pinout (More details available on datasheet)

On the right: USB to UART programmer, with access to 5v, GND and your Serial Pins

Notes and Debugging:

- Make sure your MCU voltage and programming logic level match
- Be mindful of USB common grounds and current limits (especially for 3.3v)
- BAUD Rate set at 9600 in example code
- Be sure to connect VDD & AVDD for all ATMEGA use (even if not using the ADC)

Feel free to send any questions to sales@meridianelectronic.com

Check MeridianElectronic.com for datasheets, 3D printing, solder stencils & more!