ATtiny817 Xplained Mini

USER GUIDE

Preface

The Atmel® ATtiny817 Xplained Mini evaluation kit is a hardware platform to evaluate the ATtiny817 microcontroller.

Supported by the Atmel Studio integrated development platform, the kit provides easy access to the features of the Atmel ATtiny817 and explains how to integrate the device in a custom design.

The Xplained Mini series evaluation kits include an on-board mini embedded debugger, and no external tools are necessary to program the ATtiny817.

Table of Contents

1. Introduction

1.1. Features

- ATtiny817 microcontroller
- One yellow user LED
- One mechanical button
- Two QTouch® buttons
- mEDBG
    - Auto-ID for board identification in Atmel Studio
    - One green board status LED
    - Programming and Debugging
    - Virtual COM port (CDC)
- USB powered
- ATtiny817 power sources:
    - 5.0V from USB
    - 3.3V regulator
    - External voltage
- Arduino shield compatible footprints

1.2. Kit Overview

The Atmel ATtiny817 Xplained Mini evaluation kit is a hardware platform to evaluate the Atmel ATtiny817.

Figure 1-1. ATtiny817 Xplained Mini Evaluation Kit Overview
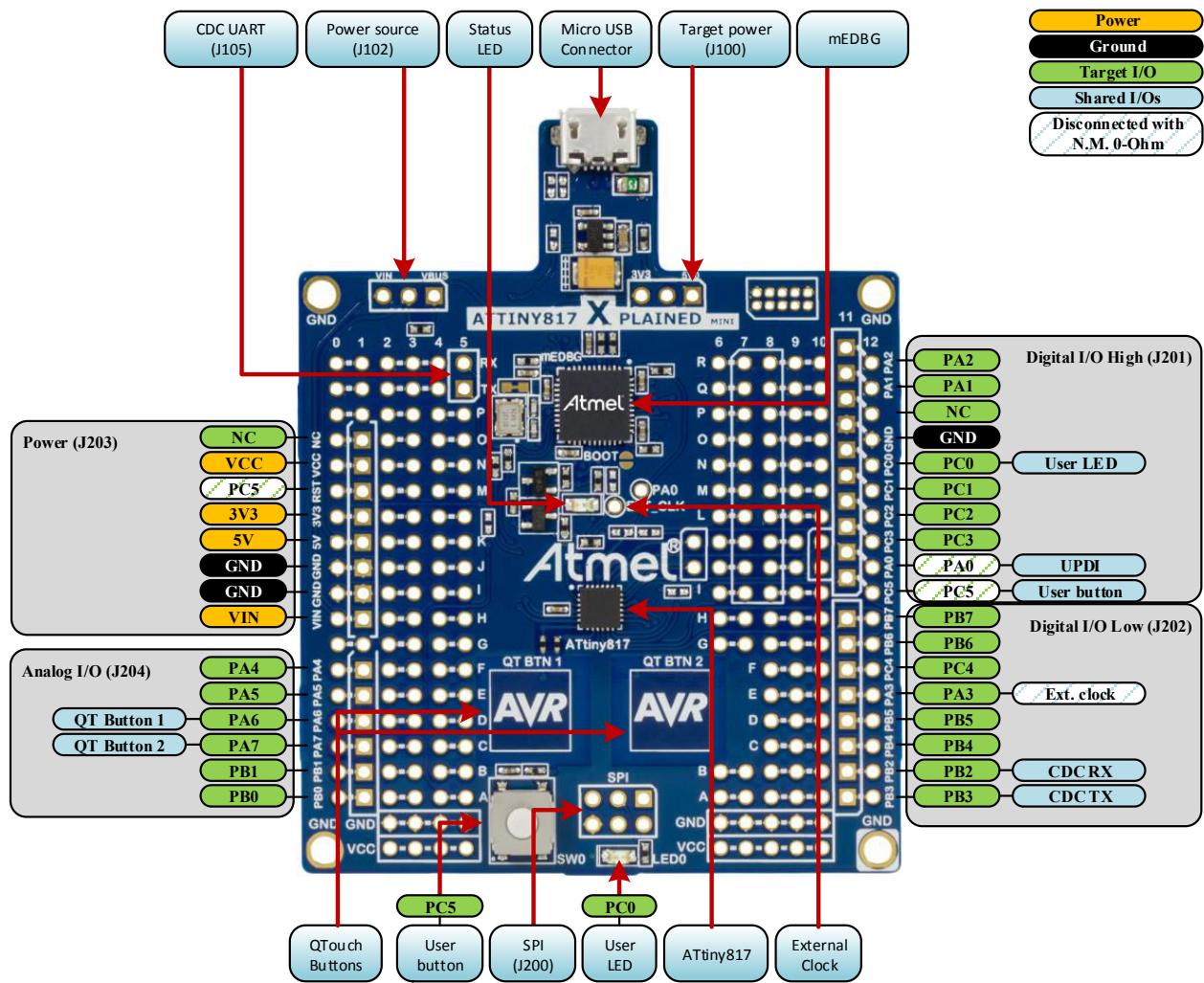
Table 1-1. Default Configurations

Function	Default configuration	Other settings
Kit power source (J102)	5.0V USB powered	External input VIN [1]
Target power (J100)	5.0V USB powered	3.3V from on-board regulator [1]
ATtiny817 clock	Internal 20MHz oscillator	mEDBG clock [2]

Info:
Changing the default settings require modification of the kit using a soldering iron.

1) Details on how to change the power settings are described in Power Sources.

2) Details on the mEDBG clock are described in Xplained Mini Clock Output.

2. Getting Started

2.1. Xplained Mini Quick Start

Steps to start exploring the Atmel Xplained Mini platform:
1. Download Atmel Studio.
2. Launch Atmel Studio.
3. Connect a USB cable (Standard-A to Micro-B or Micro-AB) between the PC and the USB port on the kit.

When the Xplained Mini kit is connected to your computer for the first time, the operating system will perform a driver software installation. The driver file supports both 32- and 64-bit versions of Microsoft® Windows® XP, Windows Vista®, Windows 7, Windows 8, and Windows 10. The drivers for the kit are included with Atmel Studio.

Once the Xplained Mini board is powered the green status LED will blink and Atmel Studio will auto detect which Xplained Mini board is connected. Atmel Studio will present relevant information like datasheets and kit documentation. The ATtiny817 device is programmed and debugged by the on-board Mini Embedded Debugger and therefore no external programmer or debugger tool is required.

2.2. Design Documentation and Relevant Links

The following list contains links to the most relevant documents and software for the ATtiny817 Xplained Mini.

- **Xplained products** - Atmel Xplained evaluation kits are a series of easy-to-use evaluation kits for Atmel microcontrollers and other Atmel products. For low pin-count devices the Xplained Nano series provides a minimalistic solution with access to all I/O pins of the target microcontroller. Xplained Mini kits are for medium pin-count devices and adds Arduino UNO compatible header footprint and a prototyping area. Xplained Pro kits are for medium to high pin-count devices, they features advanced debugging and standardized extensions for peripheral functions. All these kits have on board programmers/debuggers which creates a set of low-cost boards for evaluation and demonstration of features and capabilities of different Atmel products.
- **Atmel Studio** - Free Atmel IDE for development of C/C++ and assembler code for Atmel microcontrollers.
- **Atmel sample store** - Atmel sample store where you can order samples of devices.
- **Atmel Data Visualizer** - Atmel Data Visualizer is a program used for processing and visualizing data. Data Visualizer can receive data from various sources such as the Embedded Debugger Data Gateway Interface found on Xplained Pro boards and COM ports.
- **Atmel QTouch® Library PTC** - QTouch Library for Atmel AVR® and ARM®-based microcontrollers.
- **Atmel QTouch® Composer** - Tool for developing capacitive buttons, sliders, and wheels applications.
- **Atmel QTouch® Design Guide** - PTC Robustness design guide document for touch sensor development.
- **Design Documentation** - Package containing CAD source, schematics, BOM, assembly drawings, 3D plots, layer plots, etc.
- **Hardware Users Guide in PDF format** - PDF version of this User Guide.

3. Xplained Mini

Xplained Mini is an evaluation platform that provides a set of small boards with access to all microcontoller I/O's. The platform consists of a series of low pin-count Microcontroller (MCU) boards, which are integrated with Atmel Studio to present relevant user guides, application notes, datasheets, and example code. The platform also features a Virtual COM port for serial communication to a host PC.

3.1. Mini Embedded Debugger

The ATtiny817 Xplained Mini contains the Atmel Mini Embedded Debugger (mEDBG) for on-board programming and debugging. The mEDBG is a composite USB device of two interfaces; a debugger and a Virtual COM Port.

Together with Atmel Studio, the mEDBG debugger interface can program and debug the ATtiny817. On ATtiny817 Xplained Mini, the UPDI interface is connected between the mEDBG and the ATtiny817.

The Virtual COM Port is connected to a UART on the ATtiny817 and provides an easy way to communicate with the target application through terminal software. It offers variable baud rate, parity, and stop bit settings.
Note: The settings on the ATtiny817 must match the settings given in the terminal software.

Info: The virtual COM port in the mEDBG requires the terminal software to set the data terminal ready (DTR) signal to enable the UART pins connected to the ATtiny817. If the DTR signal is not enabled the UART pins on the mEDBG is kept in high-z (tristate) rendering the COM port unusable. The DTR signal is set automatically by some terminal software, but it may have to be manually enabled in your terminal.

The mEDBG controls one status LED on ATtiny817 Xplained Mini. The table below shows how the LED is controlled in different operation modes.

Table 3-1. mEDBG LED Control

| Operation mode	Status LED
Power up	LED is lit briefly
Normal operation	LED is not lit
Programming	Activity indicator; LED flashes when programming/debugging with the mEDBG

3.1.1. Xplained Mini Clock Output

The mEDBG outputs its CPU clock on a pin.

The clock output can be used to feed the target device with a more accurate clock if this is needed for the application. By default, this clock is disconnected from the target by a not mounted resistor.

To connect the external clock to the ATtiny817 a 0Ω resistor or strap has to be soldered in the footprint shown in the figure below (R205).

Figure 3-1. External Clock Footprint
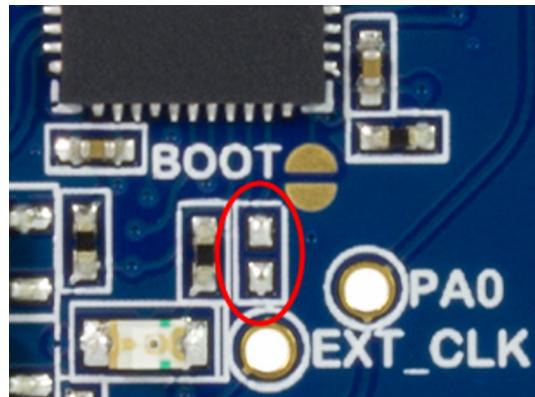
The mEDBG CPU clock frequency depends on the selected voltage and is shown in the table below.

Table 3-2. CPU Clock vs. Voltage

Target voltage	mEDBG CPU clock
3.3V	8MHz
5.0V	16MHz

3.2. mEDBG Configuration

The operation of the mEDBG can be configured by writing registers in the mEDBG. No configuration is required for default operation.

3.2.1. mEDBG Low Power Modes

There are two modes that enables the mEDBG to save power when connected to an external power source.

EOF mode where the mEDBG is disabled. When enabled the ATmega32U4 will enter sleep mode if USB does not enumerates within 5 seconds of power up. In this mode the external clock is not available to the target MCU.

LOWP mode where the mEDBG is set to run at 1MHz, saving power while maintaining the USB connection for the COM port. The external clock will be 1MHz.

Table 3-3. Low Power Modes Operation

Mode	External CLK	COM port	UPDI program	UPDI debug
EOF	disabled	disabled	disabled	disabled
LOWP	forced 1MHz	enabled	useless	useless
Factory settings	enabled	enabled	enabled	enabled

3.2.2. mEDBG Fuse Filter

The mEDBG does not initially allow users to program all fuses of the target device through Atmel Studio, a filter is implemented to protect certain fuses. The protected fuses are different for every product using the mEDBG and are typically clock related fuses that could be set to invalid configurations.

The fuse protection can be disabled by writing the FUSE bit to 0.

> **Tip:** The fuse filter prevents users from changing critical fuses using Atmel Studio, however it does not prevent users from setting fuses freely using the command line interface atprogram that is bundled with Atmel Studio.

3.2.3. mEDBG Command Line Interface

The configuration of the mEDBG can be changed using a simple command line interface available on Atmel Spaces Releases (mEDBG_script.zip).

The CLI is written for Python$^®$ 2.7 and may work on other Python 2.x versions. Python can be downloaded from Python.

The register definitions are available in the following chapters.

Related Links
SUFFER on page 9

3.2.4. Super User Fantastic Feature Enable Register

The Super User Fantastic Feature Enable Register allows the user to modify the behavior of the mEDBG.

Name: SUFFER
Offset: 0x0120
Reset: 0xFF
Property: N/A

Bit	7	6	5	4	3	2	1	0
						EOF	LOWP	FUSE
Access						R/W	R/W	R/W
Reset						1	1	1

Bit 2 – EOF: Extended Off
Writing the EOF bit to 1 sets default operation. Writing the EOF bit to 0 enables the extended off power mode. If no USB enumeration is successful within five seconds of power up, the mEDBG enters deep sleep.

Bit 1 – LOWP: Low Power
Writing the LOWP bit to 1 sets the system clock to its default value. Writing the LOWP bit to 0 enables low power mode. The mEDBG is set to run at 1MHz, which decreases power usage.

Bit 0 – FUSE: FUSE Protection
Writing the FUSE bit to 1 enables fuse protection when using Atmel Studio. The fuse protection prevents modification of specific fuses in the target device ATtiny817 that could make the mEDBG on the ATtiny817 Xplained Mini not usable. Writing the FUSE bit to 0 removes all protection of fuses in the target device ATtiny817.

> ⚠️ **Warning:** Writing the FUSE bit to 0 enables modification of all fuses in the ATtiny817. Setting wrong fuse settings may render the mEDBG not usable on the ATtiny817 Xplained Mini. As an example an invalid clock settings can be set, recovery may require an external debugger.

4. Hardware User Guide

The following sections describes the implementation of the relevant peripherals, headers, and connectors on ATtiny817 Xplained Mini and their connection to the ATtiny817. The tables of connections in the sections also describes which signals are shared between the headers and on-board functionality.

The figure below shows the assembly drawing of the ATtiny817 Xplained Mini to help identification of components.

Figure 4-1. ATtiny817 Xplained Mini Assembly Drawing
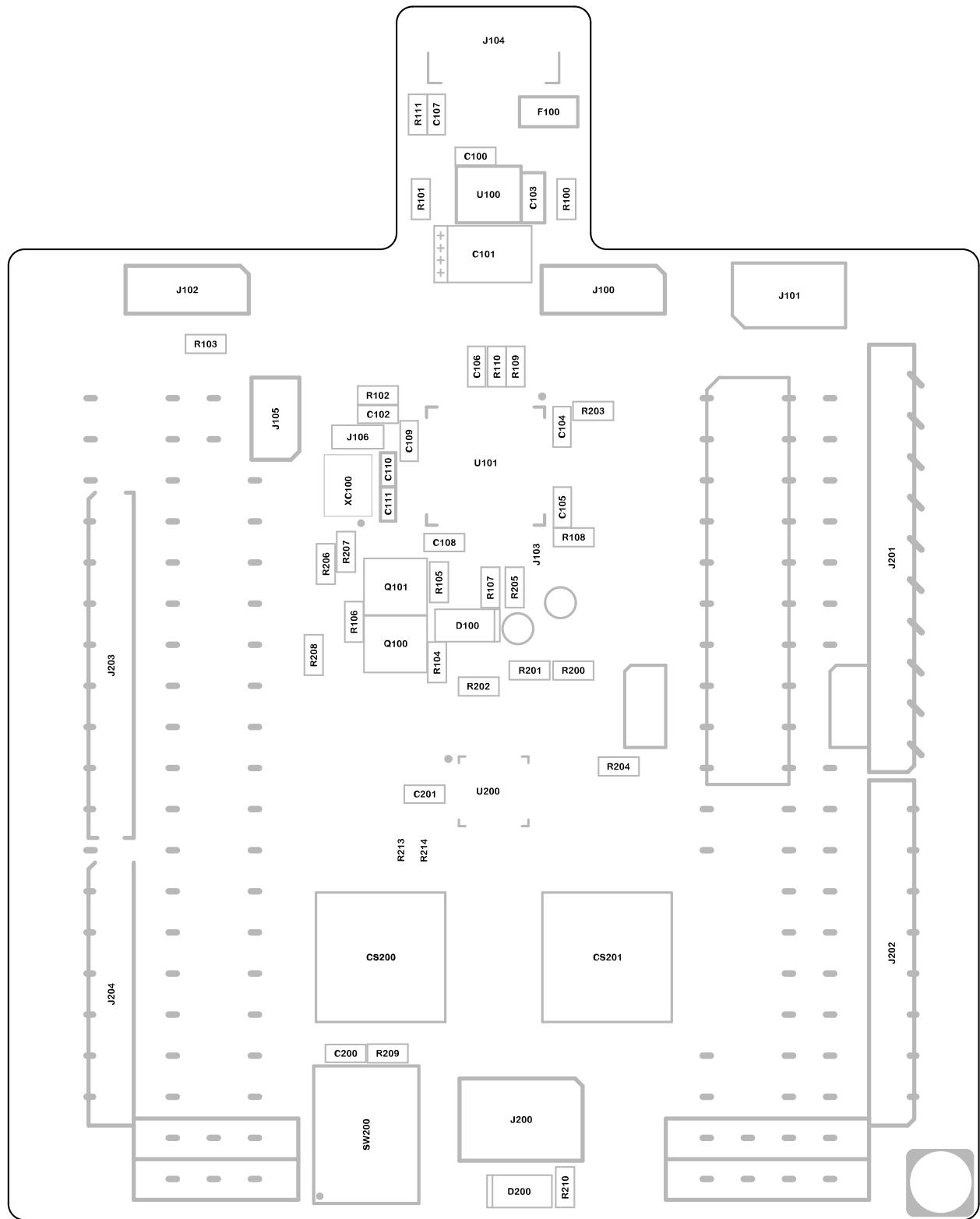
4.1. Power Sources

ATtiny817 Xplained Mini kit can be powered by USB or an external voltage input VIN. The default power source is 5.0V from USB, the USB port is protected with a 500mA PTC resettable fuse.

The ATtiny817 is powered from the 5.0V USB voltage by default.

The figure below shows the possible kit power supply connections.

Figure 4-2. Power Supply Block Diagram
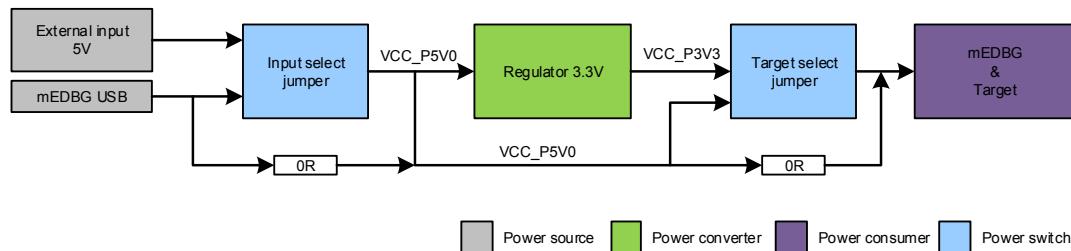
The input voltage select jumper (J102, 3-pin header footprint) can be soldered in to select between power from the USB port or from the VIN pin on the Arduino power header footprint. By default the selector is bypassed with a 0Ω resistor (R103) to connect USB voltage to the on-board 3.3V 150mA regulator and target voltage select header.

The target voltage select jumper (J100, 3-pin header footprint) can be soldered in to select between the on-board 3.3V regulator or the voltage from the input voltage select jumper (J102). The selector is bypassed with a 0Ω resistor (R100) to connect the kit input voltage to the ATtiny817 and mEDBG.

> **Caution:** If the target voltage and input voltage select headers are soldered in and used with a jumper, the bypass 0Ω resistors have to be removed to avoid contention.

4.2. Connectors

4.2.1. ATtiny817 Xplained Mini Arduino Compatible Footprints

The ATtiny817 Xplained Mini has a pin-out that is compatible with Arduino shields, which can be mounted in the marked positions (J200, J201, J202, J203, and J204).

Table 4-1. J200 ISP/SPI Header

J200	ATtiny817 pin	Arduino numbering	Functions	Comment
1	PC1	MISO	SPI MISO	Shared with J201
2		VCC	VCC_P5V0	
3	PC0	SCK	SPI SCK	Shared with J201
4	PC2	MOSI	SPI MOSI	Shared with J201
5	PC5	RESET	GPIO	Shared with J201, J203, and user button
6	GND	GND	GND	

Table 4-2. J201 Digital I/O High Header

J201	ATtiny817 pin	Arduino numbering	Functions	Comment
1	PC5	D8	TCA_W5	Not connected by default. Can be connected by adding a 0Ω resistor to R204. Shared with J200, J203, and user button.
2	PA0	D9	UPDI	Not connected by default. Can be connected by adding a 0Ω resistor to R200. Shared with UPDI.
3	PC3	D10	SPI SS	
4	PC2	D11	SPI MOSI	Shared with J200
5	PC1	D12	SPI MISO	Shared with J200
6	PC0	D13	SPI SCK	Shared with J200 and user LED
7	GND	GND		
8		AREF		
9	PA1	D14	I^{2}C SDA	
10	PA2	D15	I^{2}C SCL	

Table 4-3. J202 Digital I/O Low Header

J202	ATtiny817 pin	Arduino numbering	Functions	Comment
1	PB3	D0	UART RX	Shared with mEDBG CDC
2	PB2	D1	UART TX	Shared with mEDBG CDC
3	PB4	D2	TCA W1	
4	PB5	D3	TCA W2	
5	PA3	D4	TCA W3	
6	PC4	D5	TCA W4	
7	PB6	D6		
8	PB7	D7		

Table 4-4. J203 Power Header

J203	ATtiny817 pin	Arduino numbering	Functions	Comment
1		NC		
2	VCC	IOREF	VCC_TARGET	
3	PC5	RESET	GPIO	Disconnected by not mounted 0Ω resistor R208. Shared with J200, J201, and user button.
4		3.3V	VCC_P3V3	
5		5.0V	VCC_P5V0	
6	GND	GND		

J203	ATtiny817 pin	Arduino numbering	Functions	Comment
7	GND	GND		
8		VIN		External voltage input

Table 4-5. J204 Analog I/O Header

J204	ATtiny817 pin	Arduino numbering	Functions	Comment
1	PA4	A0	AIN04	
2	PA5	A1	AIN05	
3	PA6	A2	AIN06	Shared with QTouch Button 1
4	PA7	A3	AIN07	Shared with QTouch Button 2
5	PB1	A4	AIN10	
6	PB0	A5	AIN11	

4.2.2. Extension Header Area

The marked area on the grid I7 to R8 can be used for strapping in an Xplained Pro extension header or a 10-pin legacy Xplained/RZ600 header.

Extension header area

SPI_SS
SPI_MISO

SPI_MOSI
SPI_SCK
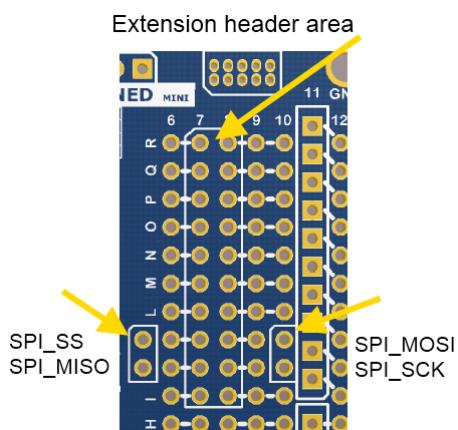
The SPI bus signals are available close to the header at row J and K, enabling easy connection to header pin 15 to 18.

Using Pin 11 to 20 enables connection of the 10-pin legacy header used on the RZ600 wireless modules and the 10-pin Xplained sensor modules.

The general bus connections for an Xplained Pro Extension board is indicated in the table below, detailed wiring can be found in the selected extension board documentation.

Table 4-6. Extension Header Typical Signals

Pin	Signal name	Signal description
1	ID	Communication line to the ID chip on the Xplained extension board
2	GND	Ground
3	ADC(+)	Analog to digital converter, alternatively positive part of differential ADC

Pin	Signal name	Signal description
4	ADC(-)	Analog to digital converter, alternatively negative part of differential ADC
5	GPIO1	General purpose I/O
6	GPIO2	General purpose I/O
7	PWM(+)	Pulse width modulation, alternatively positive part of differential PWM
8	PWM(-)	Pulse width modulation, alternatively negative part of differential PWM
9	IRQ/GPIO	Interrupt request line and/or general purpose I/O
10	SPI_SS_B/ GPIO	Slave B select for SPI and/or general purpose I/O
11	I2C_SDA	Data line for I^{2}C interface
12	I2C_SCL	Clock line for I^{2}C interface
13	UART_RX	Receiver line of ATtiny817 USART
14	UART_TX	Transmitter line of ATtiny817 USART
15	SPI_SS_A	Slave A select for SPI
16	SPI_MOSI	Master out slave in line of serial peripheral interface
17	SPI_MISO	Master in slave out line of serial peripheral interface
18	SPI_SCK	Clock for serial peripheral interface
19	GND	Ground
20	VCC	Power for extension board

4.3. Current Measurement

The power to the target controller ATtiny817 and its peripherals is connected from the *VCC_BOARD* supply to the targets *VCC_TARGET* supply with a 0Ω resistor (*R104*) as shown in the figure below. To measure the power consumption of the device, remove the 0Ω resistor and replace it with an ammeter. The ammeter can be connected between the *VCC_BOARD* (pin 2 on the target voltage select header) and *VCC_TARGET* (pin 2 on the power header) pads for easy measurement.

Caution: Removing the resistor while the kit is powered without an ammeter or 0Ω resistor mounted may cause the ATtiny817 to be powered through its I/O pins. This may cause permanent damage to the device.

Figure 4-3. Current Measurement
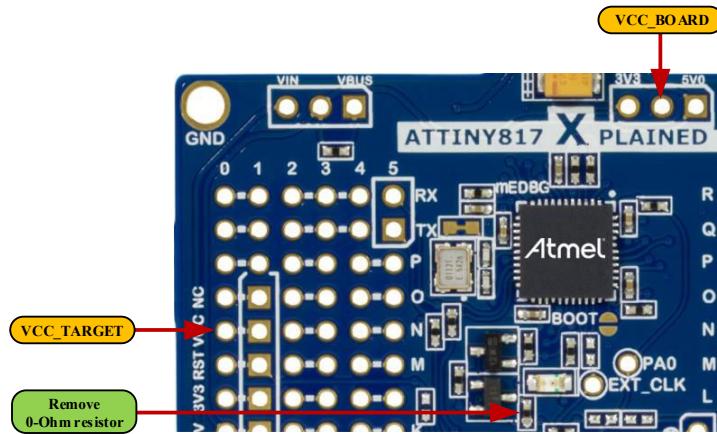
4.4.   Disconnecting mEDBG

The target controller ATtiny817 can be completely separated from the mEDBG, but this requires some small modifications to the board using a soldering iron. By removing the resistors in the sections shown in the figure below, the mEDBG is completely disconnected from the target controller. To reconnect the mEDBG again, solder in 0Ω resistors or solder in 100-mil headers on the header footprints and use wire-straps to connect the interfaces.

Figure 4-4. Kit Modifications
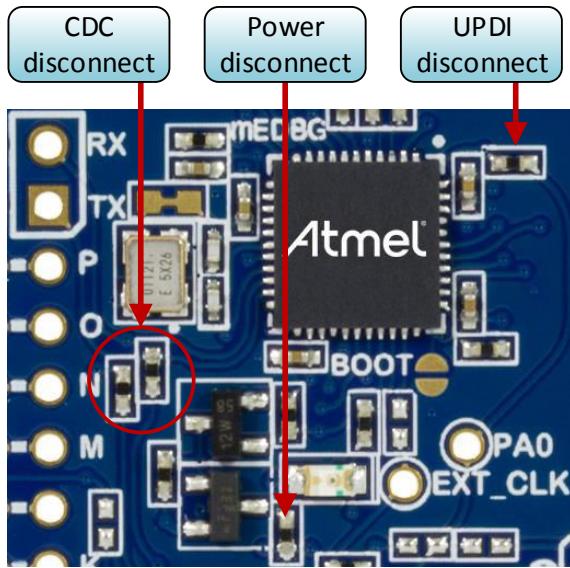
Table 4-7. mEDBG Connections

Designator	Mounted	From (mEDBG)	To (ATtiny817)
R104	Yes	VCC_BOARD	VCC_TARGET
R203	Yes	UPDI	PA0 UPDI / RESET
R205	**No**	CLK_OUT	PA3 CLKI
R206	Yes	CDC UART RX	PB2 UART TX
R207	Yes	CDC UART TX	PB3 UART RX

4.5. Peripherals

4.5.1. LED

There is one yellow LED available on the ATtiny817 Xplained Mini board that can be turned ON and OFF. The LED can be activated by driving the connected I/O line to VCC.

Table 4-8. LED Connection

ATtiny817 pin	Function	Shared functionality
PC0	Yellow LED0	Shared with J200 and J201

4.5.2. Mechanical Button

ATtiny817 Xplained Mini contains one mechanical button. This is a generic user configurable button and when a button is pressed it will drive the I/O line to GND.

> **Info:** There is an external pull-up resistor connected to the generic user button.

Table 4-9. Mechanical Button

ATtiny817 pin	Description	Shared functionality
PC5	User button	J200, J201, and J203

4.5.3. QTouch Buttons

ATtiny817 Xplained Mini contains two QTouch buttons for use with the internal PTC module of the device.

> **Tip:** The touch button on the kit is placed in the inner layers of the PCB, and has a very small overlay. Due to the short distance from the sensor to the touch area, it might be oversensitive. Different overlays can be used to avoid saturation of the sensor.

Table 4-10. QTouch Buttons

ATtiny817 pin	Function	Description	Shared functionality
PA6	PTC2	QTBTN1	J204
PA7	PTC3	QTBTN2	J204

4.6. Embedded Debugger Implementation

ATtiny817 Xplained Mini contains a Mini Embedded Debugger (mEDBG) that can be used to program the ATtiny817 using Unified Program and Debug Interface (UPDI). The mEDBG also include a Virtual Com port interface over UART. Atmel Studio can be used as a front end for the Mini Embedded Debugger.

4.6.1. Unified Program and Debug Interface

The Unified Program and Debug Interface (UPDI) uses one pin to communicate with the target.

Table 4-11. UPDI Connections

ATtiny817 pin	Function	Shared functionality
PA0	UPDI program/debug	mEDBG and J201 (N.M 0-ohm)

4.6.2. Virtual COM Port

The Embedded Debugger acts as a Virtual Com Port gateway by using one of the ATtiny817 UARTs.

Table 4-12. Virtual COM Port Connections

ATtiny817 pin	Function	Shared functionality
PB2	UART TXD (ATtiny817 TX line)	mEDBG CDC RX and J202
PB3	UART RXD (ATtiny817 RX line)	mEDBG CDC TX and J202

5. Hardware Revision History and Known Issues

This user guide is written to reflect the latest available revision of the kit. This chapter contains information about known issues, a revision history of older revisions, and how older revisions differ from the latest revision.

5.1. Identifying Product ID and Revision

The revision and product identifier of Xplained Mini boards can be found in two ways; either through Atmel Studio or by looking at the sticker on the bottom side of the PCB.

By connecting an Xplained Mini board to a computer with Atmel Studio running, an information window will pop up. The first six digits of the serial number, which is listed under kit details, contain the product identifier and revision.

The same information can be found on the sticker on the bottom side of the PCB. Most kits will print the identifier and revision in plain text as A09-nnnn\rr, where nnnn is the identifier and rr is the revision. Boards with limited space have a sticker with only a data matrix code, which contains a serial number string.

The serial number string has the following format:

> "nnnnrrssssssssss"
>
> n = product identifier
>
> r = revision
>
> s = serial number

The product identifier for ATtiny817 Xplained Mini is A09-2658.

5.2. Revision 4

Revision 4 is the initially released revision.

6. Document Revision History

Doc. rev.	Date	Comment
42726A	10/2016	Initial document release.

7. Evaluation Board/Kit Important Notice

This evaluation board/kit is intended for use for **FURTHER ENGINEERING, DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY.** It is not a finished product and may not (yet) comply with some or any technical or legal requirements that are applicable to finished products, including, without limitation, directives regarding electromagnetic compatibility, recycling (WEEE), FCC, CE or UL (except as may be otherwise noted on the board/kit). Atmel supplied this board/kit "AS IS", without any warranties, with all faults, at the buyer's and further users' sole risk. The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies Atmel from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge and any other technical or legal concerns.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER USER NOR ATMEL SHALL BE LIABLE TO EACH OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

No license is granted under any patent right or other intellectual property right of Atmel covering or relating to any machine, process, or combination in which such Atmel products or services might be or are used.

Mailing Address:

```
Atmel Corporation
1600 Technology Drive
San Jose, CA 95110
USA
```

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

[Atmel](#):
  [ATTINY817-XMINI](#)