Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ATtiny3217/ATtiny1617

8-bit tinyAVR® 1-Series Microcontroller

Introduction

The ATtiny3217/ATtiny1617 are members of the tinyAVR® 1-series of microcontrollers, using the AVR® 8-bit processor with hardware multiplier, running at up to 20 MHz and with 16 KB or 32 KB Flash, 2 KB of SRAM, and 256 bytes of EEPROM in a 24-pin package. The tinyAVR 1-series uses the latest technologies with a flexible and low-power architecture including Event System and SleepWalking, accurate analog features and advanced peripherals. Capacitive touch interfaces with proximity sensing and driven shield are supported with the integrated QTouch® peripheral touch controller.

Features

- CPU:
  - AVR® 8-bit CPU
  - Running at up to 20 MHz
  - Single-cycle I/O access
  - Two-level interrupt controller
  - Two-cycle hardware multiplier
- Memories:
  - 32/16 KB In-system self-programmable Flash memory
  - 256 bytes EEPROM
  - 2 KB SRAM
  - Write/erase endurance:
    - Flash 10,000 cycles
    - EEPROM 100,000 cycles
  - Data retention: 20 years at 85°C
- System:
  - Power-on Reset (POR)
  - Brown-out Detection (BOD)
  - Internal and external clock options:
    - 16/20 MHz low-power RC oscillator
    - 32.768 kHz Ultra Low-Power (ULP) internal RC oscillator with ±10% accuracy, ±2% calibration step size
    - 32.768 kHz external crystal oscillator
    - External clock input
  - Single pin programming and debugging interface (UPDI)
  - Three Sleep modes:
    - Idle with all peripherals running for immediate wake-up

    - Standby
        - Configurable operation of selected peripherals
        - SleepWalking peripherals
    - Power-down with limited wake-up functionality
- Peripherals:
    - One 16-bit timer/counter type A with dedicated period register, three compare channels (TCA)
    - Two 16-bit timer/counter type B with input capture (TCB)
    - One 12-bit timer/counter type D optimized for control applications (TCD)
    - 16-bit Real-Time Counter (RTC) running from an external crystal, external clock, or internal RC oscillator
    - One USART with fractional baud rate generator, auto-baud, and start-of-frame detection
    - Master/slave Serial Peripheral Interface (SPI)
    - Master/slave I^2C with dual address match
        - Standard mode (Sm, 100 kHz)
        - Fast mode (Fm, 400 kHz)
        - Fast mode plus (Fm+, 1 MHz)
    - Configurable Custom Logic (CCL) with two programmable Look-up Tables (LUT)
    - Three Analog Comparators (AC) with low propagation delay
    - Two 10-bit 115 ksps Analog-to-Digital Converters (ADC)
    - Three 8-bit Digital-to-Analog Converters (DAC) with one external channel
    - Five selectable internal voltage references: 0.55V, 1.1V, 1.5V, 2.5V, and 4.3V
    - Automated CRC memory scan
    - Window Watchdog Timer (WDT) with separate on-chip oscillator
    - Peripheral Touch Controller (PTC)
        - Capacitive touch buttons, sliders, and wheels
        - Wake-up on touch
        - Driven shield for improved moisture and noise handling performance
        - Up to 14 self-capacitance and up to 49 mutual capacitance channels
    - External interrupt on all general purpose pins
- I/O and Packages:
    - 22 programmable I/O lines
    - 24-pin QFN 4 mm x 4 mm
- Temperature Ranges:
    - -40°C to 105°C
    - -40°C to 125°C
- Speed Grades:
    - 0-5 MHz @ 1.8V – 5.5V
    - 0-10 MHz @ 2.7V – 5.5V
    - 0-20 MHz @ 4.5V – 5.5V

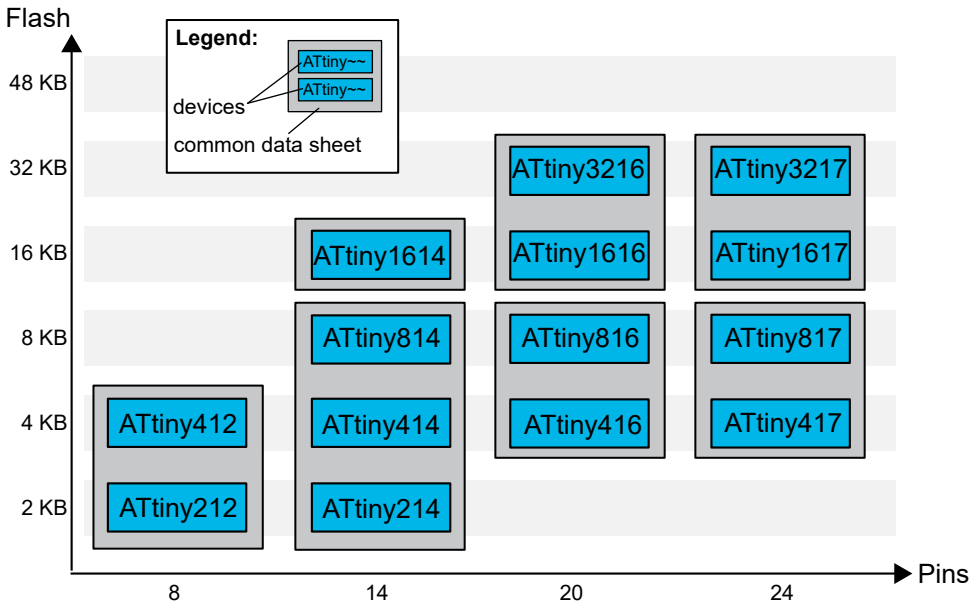
Table of Contents

1. tinyAVR® 1-series Overview

The figure below shows the tinyAVR® 1-series devices, laying out pin count variants and memory sizes:

- Vertical migration upwards is possible without code modification, as these devices are pin compatible and provide the same or more features. Downward migration may require code modification due to fewer available instances of some peripherals.
- Horizontal migration to the left reduces the pin count and therefore, the available features.

Figure 1-1. tinyAVR® 1-series Overview



Devices with different Flash memory size typically also have different SRAM and EEPROM.

Related Links

6. Memories

1.1 Configuration Summary

1.1.1 Peripheral Summary

Table 1-1. Peripheral Summary

	ATtiny1617	ATtiny3217
Pins	24	24
SRAM	2 KB	2 KB
Flash	16 KB	32 KB
EEPROM	256B	256B
Max. frequency (MHz)	20	20
16-bit Timer/Counter type A (TCA)	1	1
16-bit Timer/Counter type B (TCB)	2	2
12-bit Timer/Counter type D (TCD)	1	1
Real Time Counter (RTC)	1	1

	ATtiny1617	ATtiny3217
USART	1	1
SPI	1	1
TWI (I^2C)	1	1
ADC	2	2
ADC channels	12+12	12+12
DAC	3	3
AC	3	3
Peripheral Touch Controller (PTC)(1)	1	1
PTC number of self-capacitance channels(1)	14XY	14XY
PTC number of mutual-capacitance channels(1)	49	49
Custom Logic/Configurable Lookup Tables	1	1
Window Watchdog	1	1
Event System channels	6	6
General purpose I/O	22	22
External interrupts	22	22
CRCSCAN	1	1

Note:

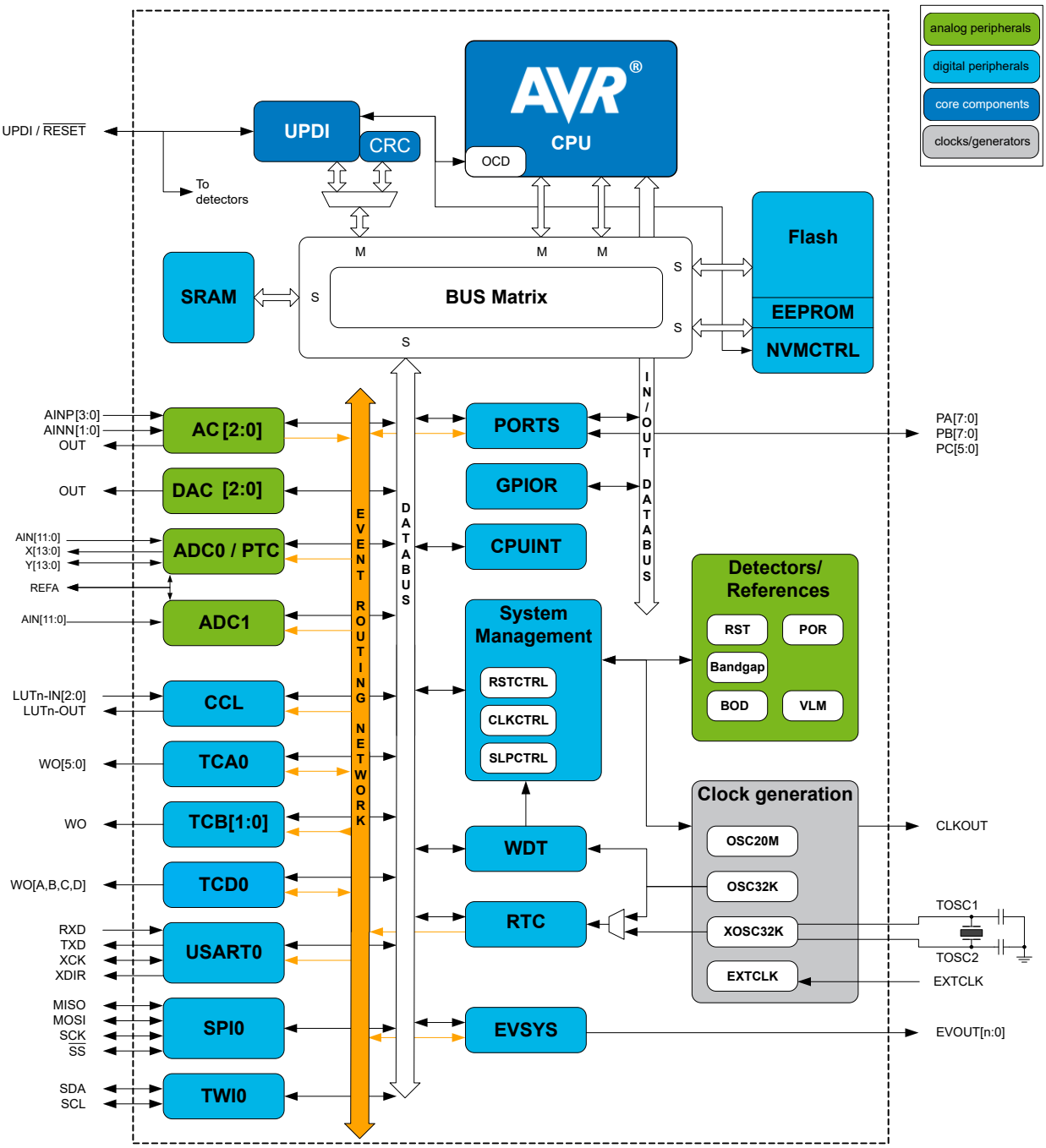
1. The PTC takes control over the ADC0 while the PTC is used.

2. Ordering Information

Find available ordering options online at microchipdirect.com, or contact your local sales representative.

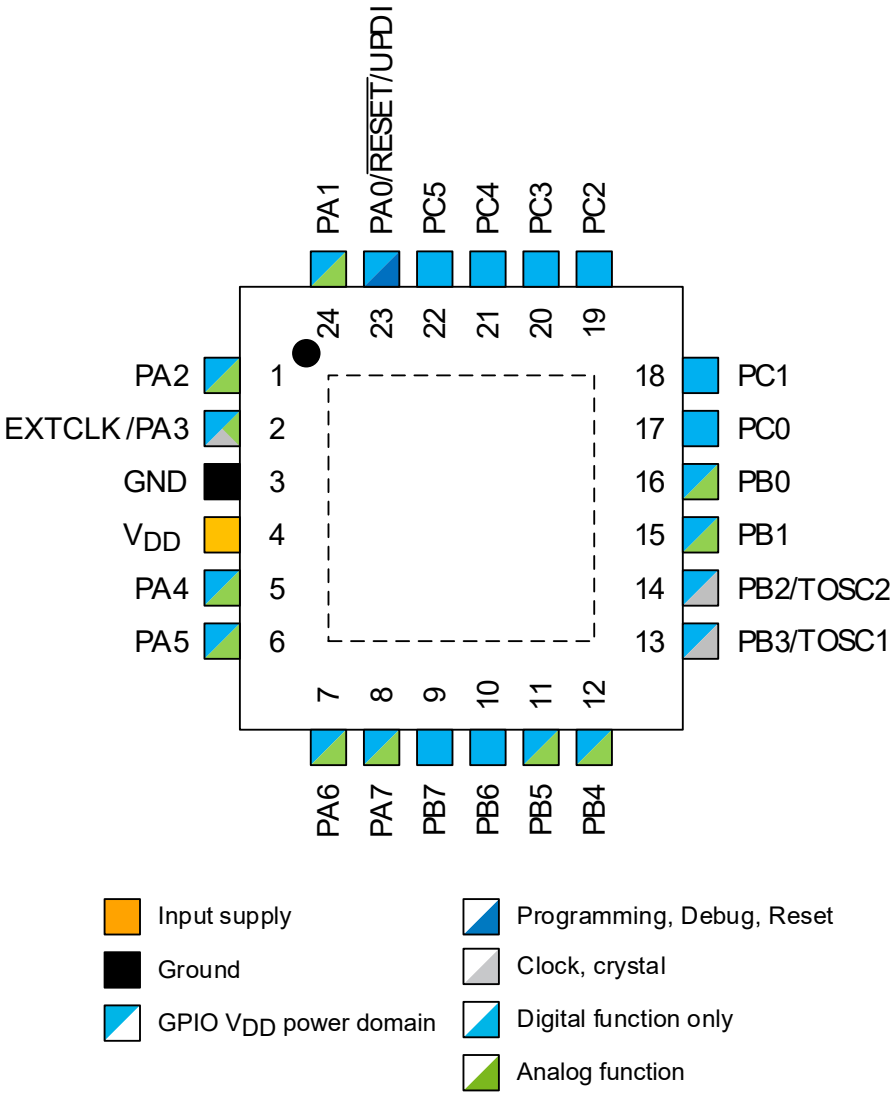
3. Block Diagram

Figure 3-1. ATtiny3217/ATtiny1617 Block Diagram



4. Pinout

4.1 24-Pin VQFN



5. I/O Multiplexing and Considerations

5.1 Multiplexed Signals

Table 5-1. PORT Function Multiplexing

VQFN 24-pin	Pin Name (1,2)	Other/Special	ADC0	ADC1	PTC(3)	AC0	AC1	AC2	DAC0	USART0	SPI0	TWI0	TCA0	TCBn	TCD0	CCL
23	PA0	$\overline{\text{RESET}}$ UPDI	AIN0													LUT0-IN0
24	PA1		AIN1							TXD	MOSI	SDA				LUT0-IN1
1	PA2	EVOUT0	AIN2							RxD	MISO	SCL				LUT0-IN2
2	PA3	EXTCLK	AIN3							XCK	SCK		WO3	TCB1 WO		
3	GND															
4	VDD															
5	PA4		AIN4	AIN0	X0/Y0					XDIR	$\overline{\text{SS}}$		WO4		WOA	LUT0-OUT
6	PA5	VREFA	AIN5	AIN1	X1/Y1	OUT	AINN0						WO5	TCB0 WO	WOB	
7	PA6		AIN6	AIN2	X2/Y2	AINN0	AINP1	AINP0	OUT							
8	PA7		AIN7	AIN3	X3/Y3	AINP0	AINP0	AINN0								LUT1-OUT
9	PB7			AIN4			AINN1	AINP3								
10	PB6			AIN5		AINP3		AINN1								
11	PB5	CLKOUT	AIN8		X12/Y12	AINP1		AINP2					WO2			
12	PB4		AIN9		X13/Y13	AINN1	AINP3						WO1			LUT0-OUT
13	PB3	TOSC1					OUT			RxD			WO0			
14	PB2	TOSC2, EVOUT1						OUT		TxD			WO2			
15	PB1		AIN10		X4/Y4	AINP2				XCK		SDA	WO1			
16	PB0		AIN11		X5/Y5		AINP2	AINP1		XDIR		SCL	WO0			
17	PC0			AIN6	X6/Y6						SCK			TCB0 WO	WOC	
18	PC1			AIN7	X7/Y7						MISO				WOD	LUT1-OUT
19	PC2	EVOUT2		AIN8	X9/Y8						MOSI					
20	PC3			AIN9	X9/Y9						$\overline{\text{SS}}$		WO3			LUT1-IN0
21	PC4			AIN10	X10/Y10								WO4	TCB1 WO		LUT1-IN1
22	PC5			AIN11	X11/Y11								WO5			LUT1-IN2

Note:

1. Pin names are of type P*xn*, with *x* being the PORT instance (A, B) and *n* the pin number. Notation for signals is PORT*x*_PIN*n*. All pins can be used as event input.
2. All pins can be used for external interrupt, where pins P*x*2 and P*x*6 of each port have full asynchronous detection.
3. Every PTC line can be configured as X- or Y-line.

Tip: Signals on alternative pin locations are in `typewriter` font.

6. Memories

6.1 Overview

The main memories are SRAM data memory, EEPROM data memory, and Flash program memory. In addition, the peripheral registers are located in the I/O memory space.

Table 6-1. Physical Properties of EEPROM

Property	ATtiny1617	ATtiny3217
Size	256 bytes	256 bytes
Page size	32 bytes	64 bytes
Number of pages	8	4
Start address	0x1400	0x1400

Table 6-2. Physical Properties of SRAM

Property	ATtiny1617	ATtiny3217
Size	2 KB	2 KB
Start address	0x3800	0x3800

Table 6-3. Physical Properties of Flash Memory

Property	ATtiny1617	ATtiny3217
Size	16 KB	32 KB
Page size	64 bytes	128 bytes
Number of pages	256	256
Start address	0x8000	0x8000

Related Links

6.2 Memory Map
6.5 EEPROM Data Memory
6.4 SRAM Data Memory
6.3 In-System Reprogrammable Flash Program Memory
9. NVMCTRL - Nonvolatile Memory Controller

6.2 Memory Map

Figure 6-1. Memory Map ATtiny3217

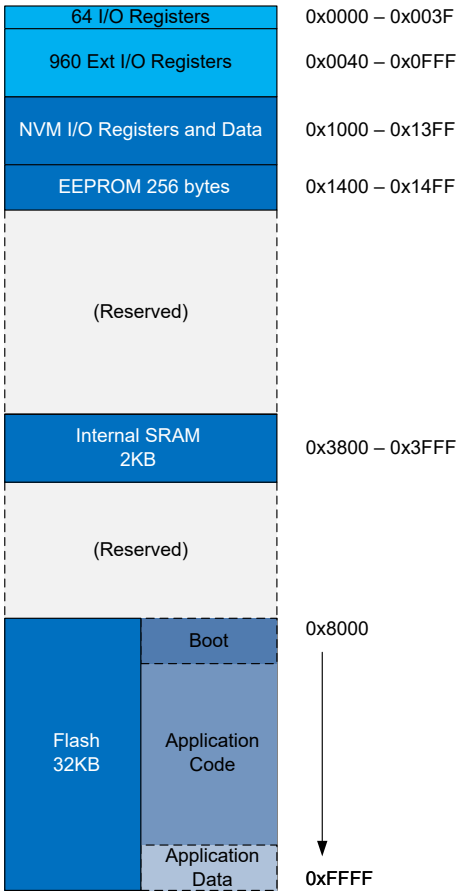
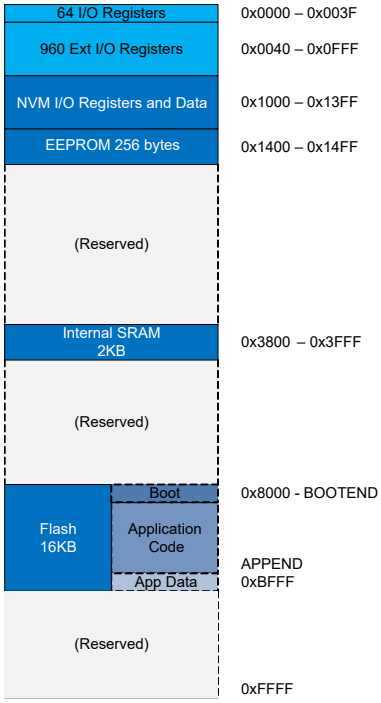


Figure 6-2. Memory Map ATtiny1617



6.3 In-System Reprogrammable Flash Program Memory

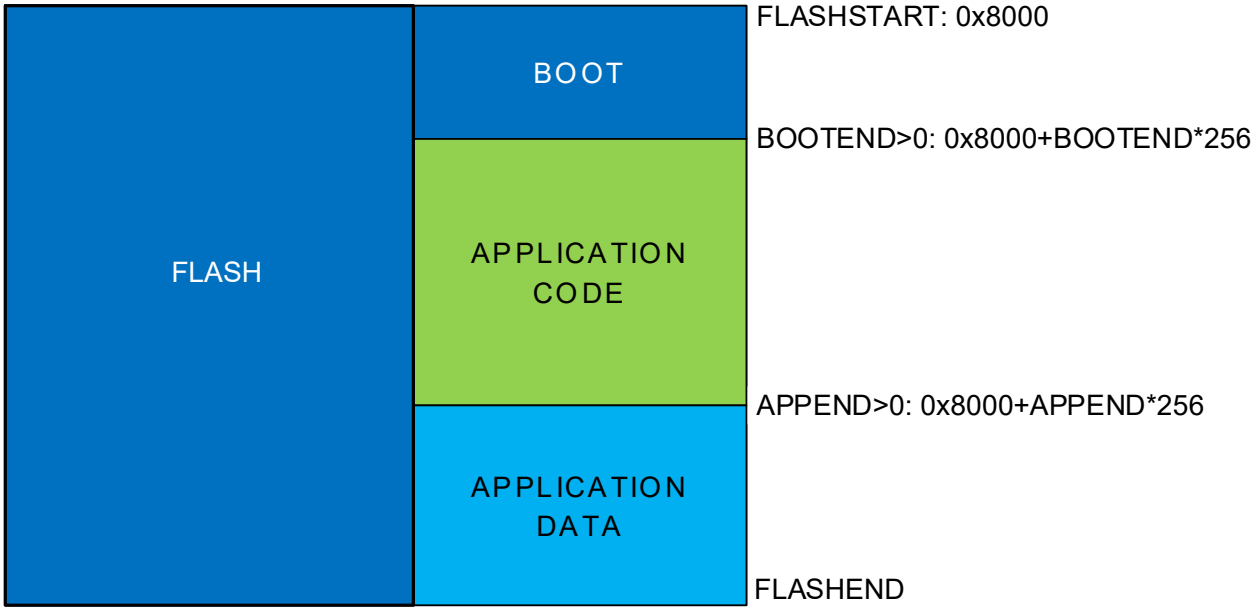
The ATtiny3217/ATtiny1617 contains 32/16 KB on-chip in-system reprogrammable Flash memory for program storage. Since all AVR instructions are 16 or 32 bits wide, the Flash is organized as 4K x 16. For write protection, the Flash program memory space can be divided into three sections (see the illustration below): Bootloader section, application code section, and application data section, with restricted access rights among them.

The Program Counter (PC) is 13/14 bits wide to address the whole program memory. The procedure for writing Flash memory is described in detail in the documentation of the Nonvolatile Memory Controller (NVMCTRL) peripheral.

The entire Flash memory is mapped in the memory space and is accessible with normal LD/ST instructions as well as the LPM instruction. For LD/ST instructions, the Flash is mapped from address 0x8000. For the LPM instruction, the Flash start address is 0x0000.

The ATtiny3217/ATtiny1617 also has a CRC peripheral that is a master on the bus.

Figure 6-3. Flash and the Three Sections



Related Links

1.1 Configuration Summary

9. NVMCTRL - Nonvolatile Memory Controller

6.4 SRAM Data Memory

The 2 KB SRAM is used for data storage and stack.

Related Links

8. AVR CPU

8.5.4 Stack and Stack Pointer

6.5 EEPROM Data Memory

The ATtiny3217/ATtiny1617 has 256 bytes of EEPROM data memory, see Memory Map section. The EEPROM memory supports single byte read and write. The EEPROM is controlled by the Nonvolatile Memory Controller (NVMCTRL).

Related Links

6.2 Memory Map

9. NVMCTRL - Nonvolatile Memory Controller

17. BOD - Brown-out Detector

6.6 User Row

In addition to the EEPROM, the ATtiny3217/ATtiny1617 has one extra page of EEPROM memory that can be used for firmware settings, the User Row (USERROW). This memory supports single byte read and write as the normal EEPROM. The CPU can write and read this memory as normal EEPROM and the UPDI can write and read it as a normal EEPROM memory if the part is unlocked. The User Row can be written by the UPDI when the part is locked. USERROW is not affected by a chip erase.

Related Links

6.2 Memory Map

9. NVMCTRL - Nonvolatile Memory Controller

33. UPDI - Unified Program and Debug Interface

6.7 Signature Bytes

All ATtiny microcontrollers have a 3-byte signature code that identifies the device. The three bytes reside in a separate address space. For the device, the signature bytes are given in the following table.

Note: When the device is locked, only the System Information Block (SIB) can be obtained.

Table 6-4. Device ID

Device Name	Signature Bytes Address		
	0x00	0x01	0x02
ATtiny1617	0x1E	0x94	0x20
ATtiny3217	0x1E	0x95	0x22

Related Links

33.3.6 System Information Block

6.8 I/O Memory

All ATtiny3217/ATtiny1617 I/Os and peripherals are located in the I/O memory space. The I/O address range from 0x00 to 0x3F can be accessed in a single cycle using `IN` and `OUT` instructions. The extended I/O memory space from 0x0040 - 0x0FFF can be accessed by the `LD`/`LDS`/`LDD` and `ST`/`STS`/`STD` instructions, transferring data between the 32 general purpose working registers and the I/O memory space.

I/O registers within the address range 0x00 - 0x1F are directly bit accessible using the `SBI` and `CBI` instructions. In these registers, the value of single bits can be checked by using the `SBIS` and `SBIC` instructions. Refer to the Instruction Set section for more details.

For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

Some of the interrupt flags are cleared by writing a '1' to them. On ATtiny3217/ATtiny1617 devices, the `CBI` and `SBI` instructions will only operate on the specified bit and can be used on registers containing such interrupt flags. The `CBI` and `SBI` instructions work with registers 0x00 - 0x1F only.

General Purpose I/O Registers

The ATtiny3217/ATtiny1617 devices provide four general purpose I/O registers. These registers can be used for storing any information, and they are particularly useful for storing global variables and interrupt flags. General purpose I/O registers, which reside in the address range 0x1C - 0x1F, are directly bit accessible using the `SBI`, `CBI`, `SBIS`, and `SBIC` instructions.

Related Links

6.2 Memory Map

34. Instruction Set Summary

6.9 Memory Section Access from CPU and UPDI on Locked Device

The device can be locked so that the memories cannot be read using the UPDI. The locking protects both the Flash (all BOOT, APPCODE, and APPDATA sections), SRAM, and the EEPROM including the FUSE data. This prevents successful reading of application data or code using the debugger interface. Regular memory access from within the application still is enabled.

The device is locked by writing any non-valid value to the LOCKBIT bit field in FUSE.LOCKBIT.

Table 6-5. Memory Access in Unlocked Mode (FUSE.LOCKBIT Valid)(1)

Memory Section	CPU Access		UPDI Access	
	Read	Write	Read	Write
SRAM	Yes	Yes	Yes	Yes
Registers	Yes	Yes	Yes	Yes
Flash	Yes	Yes	Yes	Yes
EEPROM	Yes	Yes	Yes	Yes
USERROW	Yes	Yes	Yes	Yes
SIGROW	Yes	No	Yes	No
Other Fuses	Yes	No	Yes	Yes

Table 6-6. Memory Access in Locked Mode (FUSE.LOCKBIT Invalid)(1)

Memory Section	CPU Access		UPDI Access	
	Read	Write	Read	Write
SRAM	Yes	Yes	No	No
Registers	Yes	Yes	No	No
Flash	Yes	Yes	No	No
EEPROM	Yes	No	No	No
USERROW	Yes	Yes	No	Yes(2)
SIGROW	Yes	No	No	No
Other Fuses	Yes	No	No	No

Note:

1. Read operations marked No in the tables may appear to be successful, but the data is corrupt. Hence, any attempt of code validation through the UPDI will fail on these memory sections.
2. In Locked mode, the USERROW can be written blindly using the fuse Write command, but the current USERROW values cannot be read out.

Important: The only way to unlock a device is a CHIPERASE, which will erase all device memories to factory default so that no application data is retained.

Related Links

6.10.3 Fuse Summary - FUSE

6.10.4.9 LOCKBIT

33. UPDI - Unified Program and Debug Interface

33.3.7 Enabling of KEY Protected Interfaces

6.10 Configuration and User Fuses (FUSE)

Fuses are part of the nonvolatile memory and hold factory calibration data and device configuration. The fuses are available from device power-up. The fuses can be read by the CPU or the UPDI, but can only be programmed or cleared by the UPDI. The configuration and calibration values stored in the fuses are written to their respective target registers at the end of the start-up sequence.

The content of the Signature Row fuses (SIGROW) is pre-programmed and cannot be altered. SIGROW holds information such as device ID, serial number, and calibration values.

The fuses for peripheral configuration (FUSE) are pre-programmed but can be altered by the user. Altered values in the configuration fuse will be effective only after a Reset.
Note: When writing the fuses write all reserved bits to '1'.

This device provides a User Row fuse area (USERROW) that can hold application data. The USERROW can be programmed on a locked device by the UPDI. This can be used for final configuration without having programming or debugging capabilities enabled.

Related Links

6.10.1 SIGROW - Signature Row Summary

6.10.3 Fuse Summary - FUSE

7.1 Peripheral Module Address Map

6.10.1 SIGROW - Signature Row Summary

Offset	Name	Bit Pos.									
0x00	DEVICEID0	7:0	DEVICEID[7:0]								
0x01	DEVICEID1	7:0	DEVICEID[7:0]								
0x02	DEVICEID2	7:0	DEVICEID[7:0]								
0x03	SERNUM0	7:0	SERNUM[7:0]								
0x04	SERNUM1	7:0	SERNUM[7:0]								
0x05	SERNUM2	7:0	SERNUM[7:0]								
0x06	SERNUM3	7:0	SERNUM[7:0]								
0x07	SERNUM4	7:0	SERNUM[7:0]								
0x08	SERNUM5	7:0	SERNUM[7:0]								
0x09	SERNUM6	7:0	SERNUM[7:0]								
0x0A	SERNUM7	7:0	SERNUM[7:0]								
0x0B	SERNUM8	7:0	SERNUM[7:0]								
0x0C	SERNUM9	7:0	SERNUM[7:0]								
0x0D ... 0x1F	Reserved										
0x20	TEMPSENSE0	7:0	TEMPSENSE[7:0]								
0x21	TEMPSENSE1	7:0	TEMPSENSE[7:0]								
0x22	OSC16ERR3V	7:0	OSC16ERR3V[7:0]								
0x23	OSC16ERR5V	7:0	OSC16ERR5V[7:0]								
0x24	OSC20ERR3V	7:0	OSC20ERR3V[7:0]								
0x25	OSC20ERR5V	7:0	OSC20ERR5V[7:0]								

6.10.2 Signature Row Description

6.10.2.1 Device ID n

Name: DEVICEIDn
Offset: 0x00 + n*0x01 [n=0..2]
Reset: [Device ID]
Property: -

Each device has a device ID identifying the device and its properties; such as memory sizes, pin count, and die revision. This can be used to identify a device and hence, the available features by software. The Device ID consists of three bytes: SIGROW.DEVICEID[2:0].

Bit	7	6	5	4	3	2	1	0
	DEVICEID[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bits 7:0 – DEVICEID[7:0] Byte n of the Device ID

6.10.2.2 Serial Number Byte n

Name: SERNUMn
Offset: 0x03 + n*0x01 [n=0..9]
Reset: [device serial number]
Property: -

Each device has an individual serial number, representing a unique ID. This can be used to identify a specific device in the field. The serial number consists of ten bytes: SIGROW.SERNUM[9:0].

Bit	7	6	5	4	3	2	1	0
	SERNUM[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bits 7:0 – SERNUM[7:0] Serial Number Byte n