
megaAVR[®] Data Sheet

Introduction

The Microchip AVR[®] ATmega8A is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8A achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

Features

- High-performance, Low-power Microchip AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16MIPS Throughput at 16MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 8KBytes of In-System Self-programmable Flash program memory
 - 512Bytes EEPROM
 - 1KByte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Microchip QTouch[®] library support
 - Capacitive touch buttons, sliders and wheels
 - Microchip QTouch and QMatrix[™] acquisition
 - Up to 64 sense channels

- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and VQFN package
 - Eight Channels 10-bit Accuracy
 - 6-channel ADC in PDIP package
 - Six Channels 10-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad VQFN
- Operating Voltages
 - 2.7 - 5.5V
 - 0 - 16MHz
- Power Consumption at 4MHz, 3V, 25°C
 - Active: 3.6mA
 - Idle Mode: 1.0mA
 - Power-down Mode: 0.5µA

Table of Contents

1	<i>Pin Configurations</i>	9
2	<i>Overview</i>	10
2.1	Block Diagram	10
2.2	Pin Descriptions	11
3	<i>Resources</i>	13
4	<i>Data Retention</i>	13
5	<i>About Code Examples</i>	13
6	<i>Capacitive touch sensing</i>	13
7	<i>AVR CPU Core</i>	14
7.1	Overview	14
7.2	Arithmetic Logic Unit – ALU	15
7.3	Status Register	15
7.4	General Purpose Register File	16
7.5	Stack Pointer	17
7.6	Instruction Execution Timing	18
7.7	Reset and Interrupt Handling	19
8	<i>AVR Memories</i>	22
8.1	Overview	22
8.2	In-System Reprogrammable Flash Program Memory	22
8.3	SRAM Data Memory	23
8.4	EEPROM Data Memory	24
8.5	I/O Memory	24
8.6	Register Description	25
9	<i>System Clock and Clock Options</i>	31
9.1	Clock Systems and their Distribution	31
9.2	Clock Sources	32
9.3	Crystal Oscillator	32
9.4	Low-frequency Crystal Oscillator	34
9.5	External RC Oscillator	34
9.6	Calibrated Internal RC Oscillator	36
9.7	External Clock	37
9.8	Timer/Counter Oscillator	37

9.9	Register Description	38
10	<i>Power Management and Sleep Modes</i>	39
10.1	Sleep Modes.....	39
10.2	Idle Mode.....	39
10.3	ADC Noise Reduction Mode.....	40
10.4	Power-down Mode.....	40
10.5	Power-save Mode.....	40
10.6	Standby Mode	41
10.7	Minimizing Power Consumption	41
10.8	Register Description	42
11	<i>System Control and Reset</i>	43
11.1	Resetting the AVR.....	43
11.2	Reset Sources	43
11.3	Internal Voltage Reference.....	46
11.4	Watchdog Timer	47
11.5	Timed Sequences for Changing the Configuration of the Watchdog Timer	47
11.6	Register Description	49
12	<i>Interrupts</i>	51
12.1	Interrupt Vectors in ATmega8A	51
12.2	Register Description	54
13	<i>I/O Ports</i>	56
13.1	Overview.....	56
13.2	Ports as General Digital I/O.....	57
13.3	Alternate Port Functions	61
13.4	Register Description	69
14	<i>External Interrupts</i>	71
14.1	Register Description	71
15	<i>8-bit Timer/Counter0</i>	73
15.1	Features	73
15.2	Overview.....	73
15.3	Timer/Counter Clock Sources	74
15.4	Counter Unit	74
15.5	Operation.....	74
15.6	Timer/Counter Timing Diagrams	75

15.7	Register Description	76
16	Timer/Counter0 and Timer/Counter1 Prescalers	78
16.1	Overview.....	78
16.2	Internal Clock Source	78
16.3	Prescaler Reset	78
16.4	External Clock Source	78
16.5	Register Description	79
17	16-bit Timer/Counter1	80
17.1	Features	80
17.2	Overview.....	80
17.3	Accessing 16-bit Registers	82
17.4	Timer/Counter Clock Sources	85
17.5	Counter Unit	85
17.6	Input Capture Unit	86
17.7	Output Compare Units.....	88
17.8	Compare Match Output Unit.....	90
17.9	Modes of Operation	91
17.10	Timer/Counter Timing Diagrams	98
17.11	Register Description	99
18	8-bit Timer/Counter2 with PWM and Asynchronous Operation	106
18.1	Features	106
18.2	Overview.....	106
18.3	Timer/Counter Clock Sources	107
18.4	Counter Unit	107
18.5	Output Compare Unit.....	108
18.6	Compare Match Output Unit.....	110
18.7	Modes of Operation	111
18.8	Timer/Counter Timing Diagrams	115
18.9	Asynchronous Operation of the Timer/Counter	116
18.10	Timer/Counter Prescaler	118
18.11	Register Description	119
19	Serial Peripheral Interface – SPI	123
19.1	Features	123
19.2	Overview.....	123
19.3	$\overline{SS}$ Pin Functionality	128

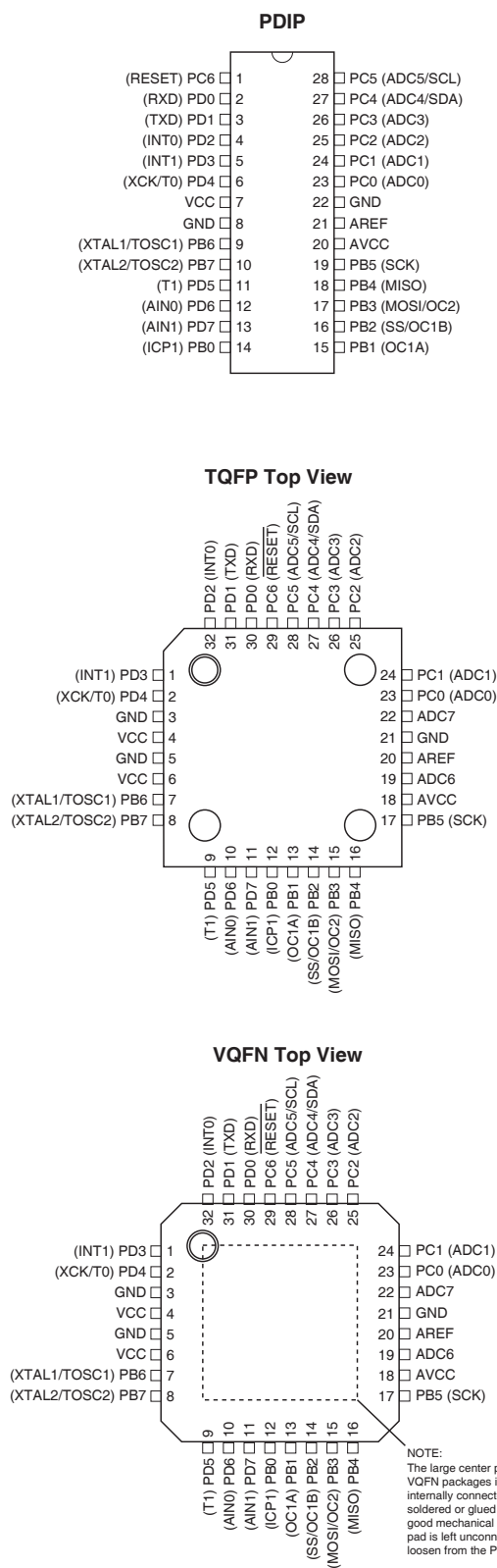
19.4	Data Modes	128
19.5	Register Description	130
20	USART	132
20.1	Features	132
20.2	Overview.....	132
20.3	Clock Generation.....	134
20.4	Frame Formats	136
20.5	USART Initialization.....	137
20.6	Data Transmission – The USART Transmitter	138
20.7	Asynchronous Data Reception	145
20.8	Multi-processor Communication Mode	148
20.9	Accessing UBRRH/UCSRC Registers	149
20.10	Register Description	150
20.11	Examples of Baud Rate Setting.....	154
21	Two-wire Serial Interface	159
21.1	Features	159
21.2	Overview.....	159
21.3	Two-wire Serial Interface Bus Definition.....	161
21.4	Data Transfer and Frame Format.....	162
21.5	Multi-master Bus Systems, Arbitration and Synchronization.....	164
21.6	Using the TWI.....	166
21.7	Multi-master Systems and Arbitration.....	181
21.8	Register Description	183
22	Analog Comparator	186
22.1	Overview.....	186
22.2	Analog Comparator Multiplexed Input	186
22.3	Register Description	187
23	Analog-to-Digital Converter	189
23.1	Features	189
23.2	Overview.....	189
23.3	Starting a Conversion	191
23.4	Prescaling and Conversion Timing.....	191
23.5	Changing Channel or Reference Selection	193
23.6	ADC Noise Canceler	194
23.7	ADC Conversion Result.....	197

23.8	Register Description	197
24	<i>Boot Loader Support – Read-While-Write Self-Programming</i>	201
24.1	Features	201
24.2	Overview.....	201
24.3	Application and Boot Loader Flash Sections.....	201
24.4	Read-While-Write and No Read-While-Write Flash Sections.....	201
24.5	Boot Loader Lock Bits	204
24.6	Entering the Boot Loader Program.....	205
24.7	Addressing the Flash During Self-Programming	205
24.8	Self-Programming the Flash.....	206
24.9	Register Description	212
25	<i>Memory Programming</i>	214
25.1	Program And Data Memory Lock Bits	214
25.2	Fuse Bits.....	215
25.3	Signature Bytes	216
25.4	Calibration Byte	216
25.5	Page Size	217
25.6	Parallel Programming Parameters, Pin Mapping, and Commands	217
25.7	Parallel Programming	219
25.8	Serial Downloading.....	227
25.9	Serial Programming Pin Mapping.....	227
26	<i>Electrical Characteristics – TA = -40°C to 85°C</i>	232
26.1	Absolute Maximum Ratings*	232
26.2	DC Characteristics.....	232
26.3	Speed Grades	234
26.4	Clock Characteristics.....	234
26.5	System and Reset Characteristics	235
26.6	Two-wire Serial Interface Characteristics	236
26.7	SPI Timing Characteristics	237
26.8	ADC Characteristics	239
27	<i>Electrical Characteristics – TA = -40°C to 105°C</i>	240
27.1	DC Characteristics.....	240
28	<i>Typical Characteristics – TA = -40°C to 85°C</i>	242
28.1	Active Supply Current.....	242
28.2	Idle Supply Current.....	246

28.3	Power-down Supply Current.....	249
28.4	Power-save Supply Current.....	250
28.5	Standby Supply Current	251
28.6	Pin Pull-up	254
28.7	Pin Driver Strength	256
28.8	Pin Thresholds and Hysteresis.....	260
28.9	Bod Thresholds and Analog Comparator Offset.....	265
28.10	Internal Oscillator Speed	267
28.11	Current Consumption of Peripheral Units.....	274
28.12	Current Consumption in Reset and Reset Pulsewidth	277
29	Typical Characteristics – $T_A = -40^{\circ}\text{C}$ to 105°C	279
29.1	ATmega8A Typical Characteristics	279
30	Register Summary	308
31	Instruction Set Summary	310
32	Ordering Information	313
33	Packaging Information	314
33.1	32A	314
33.2	28P3	315
34	Errata	317
34.1	ATmega8A, rev. L.....	317
35	Datasheet Revision History	319
35.1	Rev. B – 03/2020.....	321
35.2	Rev. A – 12/2017.....	321
35.3	Rev.8159F – 07/2015	321
35.4	Rev.8159E – 02/2013.....	319
35.5	Rev.8159D – 02/11.....	322
35.6	Rev.8159C – 07/09.....	322
35.7	Rev.8159B – 05/09.....	322
35.8	Rev.8159A – 08/08.....	323

1. Pin Configurations

Figure 1-1. Pinout

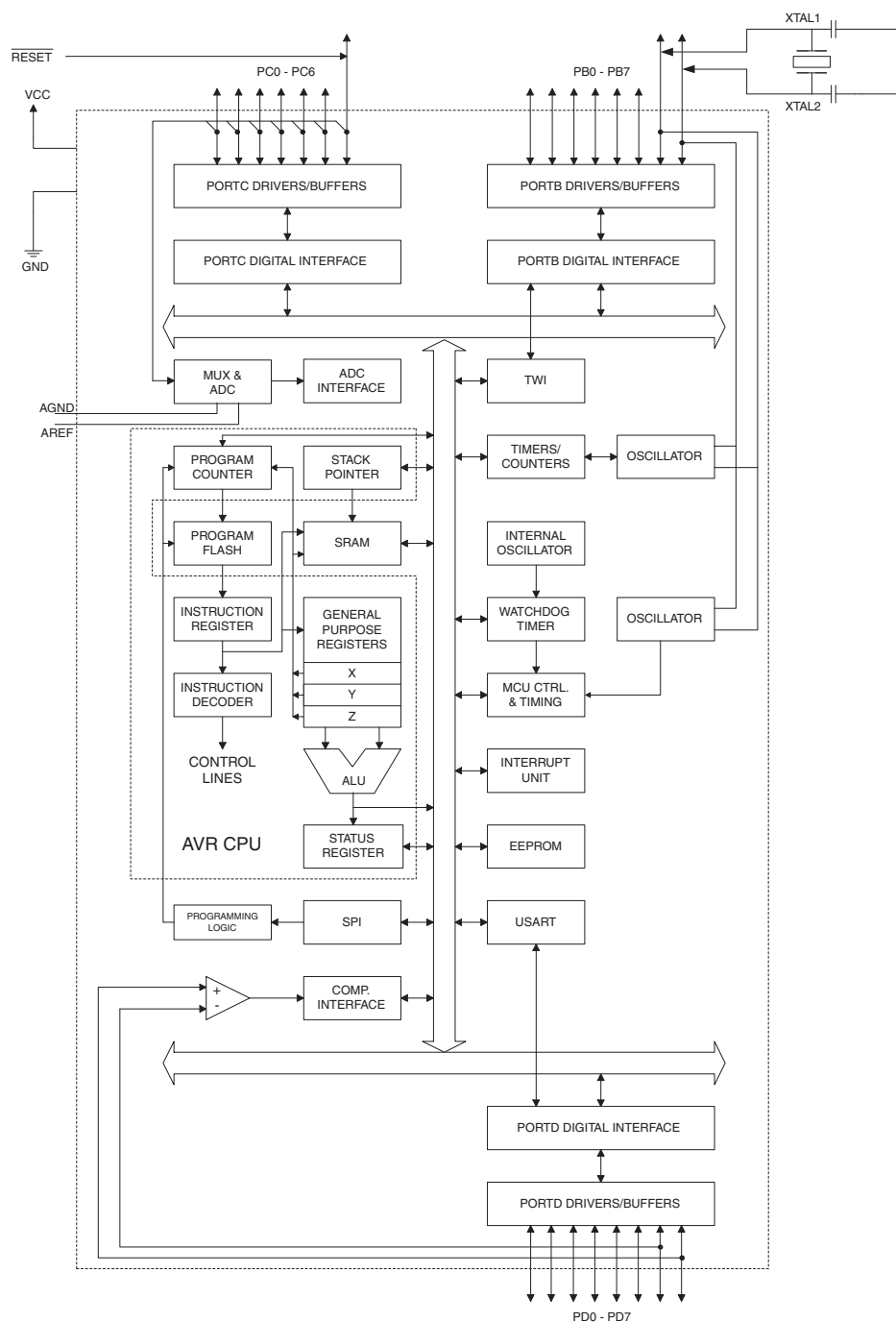


2. Overview

The Microchip AVR[®] ATmega8A is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8A achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

2.1 Block Diagram

Figure 2-1. Block Diagram



The Microchip AVR[®] core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8A provides the following features: 8K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1K byte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and VQFN packages) with 10-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Microchip's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Microchip ATmega8A is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The Microchip AVR ATmega8A is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program simulators and evaluation kits.

2.2 Pin Descriptions

2.2.1 VCC

Digital supply voltage.

2.2.2 GND

Ground.

2.2.3 Port B (PB7:PB0) – XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7:6 is used as TOSC2:1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in [“Alternate Functions of Port B” on page 63](#) and [“System Clock and Clock Options” on page 31](#).

2.2.4 Port C (PC5:PC0)

Port C is an 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

2.2.5 PC6/ $\overline{\text{RESET}}$

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 26-3 on page 235](#). Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated on [page 66](#).

2.2.6 Port D (PD7:PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega8A as listed on [page 68](#).

2.2.7 RESET

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in [Table 26-3 on page 235](#). Shorter pulses are not guaranteed to generate a reset.

2.2.8 AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, Port C (3:0), and ADC (7:6). It should be externally connected to V_{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that Port C (5:4) use digital supply voltage, V_{CC} .

2.2.9 AREF

AREF is the analog reference pin for the A/D Converter.

2.2.10 ADC7:6 (TQFP and VQFN Package Only)

In the TQFP and VQFN package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

3. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on <https://www.microchip.com/>.

4. Data Retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

5. About Code Examples

This datasheet contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.

6. Capacitive touch sensing

The Microchip QTouch[®] Library provides a simple to use solution to realize touch sensitive interfaces on most Microchip AVR[®] microcontrollers. The QTouch Library includes support for the QTouch and QMatrix[™] acquisition methods.

Touch sensing can be added to any application by linking the appropriate Microchip QTouch Library for the AVR Microcontroller. This is done by using a simple set of APIs to define the touch channels and sensors, and then calling the touch sensing API's to retrieve the channel information and determine the touch sensor states.

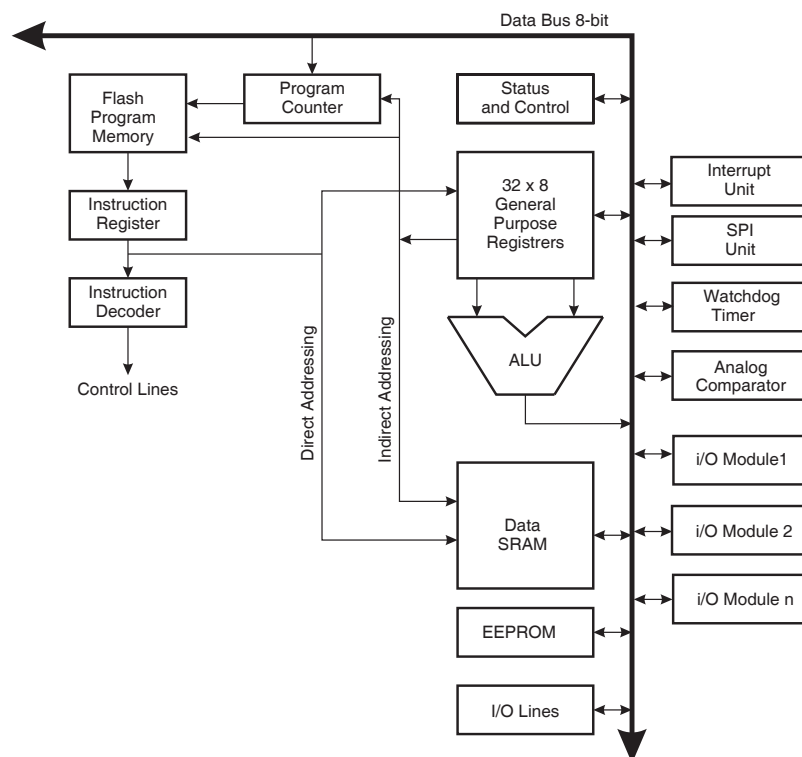
The QTouch Library is FREE and downloadable from the Microchip website at the following location: www.microchip.com. For implementation details and other information, refer to the [Atmel QTouch Library User Guide](#) - also available for download from the Microchip website.

7. AVR CPU Core

7.1 Overview

This section discusses the Microchip AVR® core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories, perform calculations, control peripherals, and handle interrupts.

Figure 7-1. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the Program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the Program memory. This concept enables instructions to be executed in every clock cycle. The Program memory is In-System Reprogrammable Flash memory.

The fast-access Register File contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-, Y-, and Z-register, described later in this section.