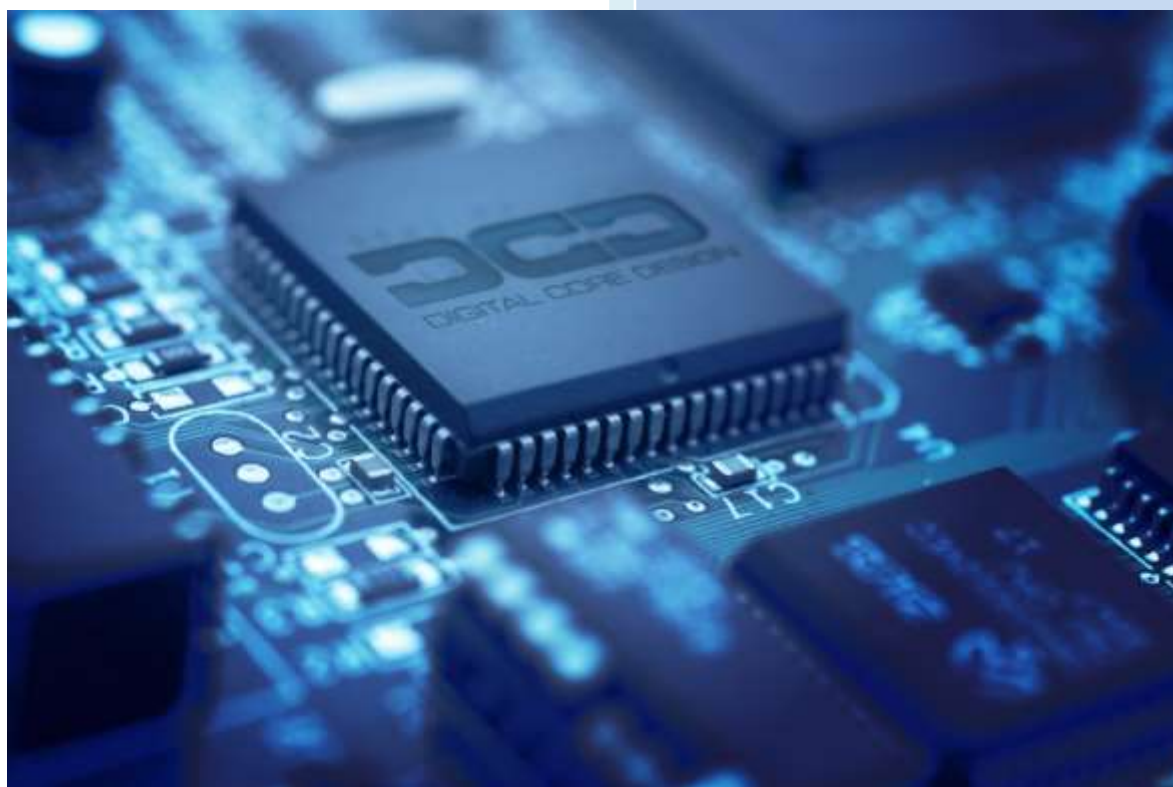




2017

# DP80390XP IP Core



Pipelined High Performance 8-bit Microcontroller v. 5.02

## COMPANY OVERVIEW

Digital Core Design is a leading IP Core provider and a System-on-Chip design house. The company was founded in 1999 and since the very beginning has been focused on IP Core architecture improvements. Our innovative, silicon proven solutions have been employed by over 300 customers and with more than 500 hundred licenses sold to companies like Intel, Siemens, Philips, General Electric, Sony and Toyota. Based on more than 70 different architectures, starting from serial interfaces to advanced micro-controllers and SoCs, we are designing solutions tailored to your needs.

## IP CORE OVERVIEW

The DP80390XP is an **ultra-high performance, speed optimized** soft core of a single-chip 8-bit embedded controller, intended to operate with **fast** (typically on-chip) and **slow** (off-chip) **memories**. It supports up to **8 MB of linear code space** and 16 MB of linear data space. The core has been designed with a special concern about **performance to power consumption ratio**. This ratio is extended by an **advanced power management (PMU) unit**. The DP80390XP soft core is 100% binary-compatible with the industry standard 8051 8-bit microcontroller. There are two configurations of the DP80390XP: **Harvard**, where internal data and program buses are separated and **von Neumann**, with common program and external data bus. The DP80390XP has a **Pipelined RISC architecture** and executes **85-200 million instructions per second**. **Dhrystone 2.1 benchmark program runs from 11.46 to 15.55 times faster, than the original 80C51 at the same frequency**. The same C compiler was used for benchmarking of the 80C51 core with the same settings. This performance can be exploited to a great advantage in **low power** applications, where the core can be clocked over ten times slower than the original implementation, without performance depletion. The DP80390XP is delivered with **fully automated test bench** and **complete set of tests**, allowing easy package validation, at each stage of SoC design flow.

## PERIPHERALS

- DoCD™ debug unit
  - Processor execution control
    - Run
    - Halt
    - Step into instruction
    - Skip instruction
  - Read-write all processor contents
    - Program Counter (PC)
    - Program Memory
    - Internal (direct) Data Memory
    - Special Function Registers (SFRs)
    - External Data Memory
  - Code execution breakpoints
    - up to eight real-time PC breakpoints
    - unlimited number of real-time OPCODE breakpoints
  - Hardware execution watch-point
    - one at Internal (direct) Data Memory
    - one at Special Function Registers (SFRs)
    - one at External Data Memory
  - Hardware watch-points activated at certain
    - address by any write into memory
    - address by any read from memory
    - address by write into memory a required data
    - address by read from memory a required data
- Unlimited number of software watch-points
  - Internal (direct) Data Memory
  - Special Function Registers (SFRs)
  - External Data Memory
- Unlimited number of software breakpoints
  - Program Memory(PC)
- Automatic adjustment of debug data transfer speed rate between HAD and Silicon
- TTAG or JTAG Communication interface
- Power Management Unit
  - Power management mode
  - Switchback feature
  - Stop mode
- Extended Interrupt Controller
  - 2 priority levels
  - Up to 7 external interrupt sources
  - Up to 8 interrupt sources from peripherals
- Four 8-bit I/O Ports
  - Bit addressable data direction for each line
  - Read/write of single line and 8-bit group
- Three 16-bit timer/counters
  - Timers clocked by internal source
  - Auto reload 8/16-bit timers
  - Externally gated event counters
- Full-duplex serial port
  - Synchronous mode, fixed baud rate
  - 8-bit asynchronous mode, fixed baud rate
  - 9-bit asynchronous mode, fixed baud rate
  - 9-bit asynchronous mode, variable baud rate
- I2C bus controller - Master
  - 7-bit and 10-bit addressing modes
  - NORMAL, FAST, FAST+, HIGH speeds
  - Multi-master systems supported
  - Clock arbitration and synchronization
  - User defined timings on I2C lines
  - Wide range of system clock frequencies
  - Interrupt generation
- I2C bus controller - Slave
  - NORMAL speed 100 kB/s
  - FAST speed 400 kB/s
  - FAST+ speed 1000 kB/s
  - HIGH speed 3400 kB/s
  - Wide range of system clock frequencies
  - User defined data setup time on I2C lines
  - Interrupt generation
- SPI – Master and Slave Serial Peripheral Interface
  - Supports speeds up ¼ of system clock
    - Mode fault error
    - Write collision error
  - Four transfer formats supported
  - System errors detection
  - Allows operation from a wide range of system clock frequencies (build-in 5-bit timer)
  - Interrupt generation
- Programmable Watchdog Timer
- 16-bit Compare/Capture Unit
  - Events capturing
  - Pulses generation
  - Digital signals generation
  - Gated timers
  - Sophisticated comparator
  - Pulse width modulation
  - Pulse width measuring
- Fixed-Point arithmetic coprocessor
  - Multiplication - 16bit \* 16bit
  - Multiplication - 32bit \* 32bit
  - Division - 32bit / 32bit
  - Division - 16bit / 16bit

- Floating-Point arithmetic coprocessor IEEE-754 standard single precision
  - *FADD, FSUB* - addition, subtraction
  - *FMUL, FDIV* - multiplication, division
  - *FSQRT* - square root
  - *FUCOM* - compare
  - *FCHS* - change sign
  - *FABS* - absolute value
- Floating-Point math coprocessor - IEEE-754 standard single precision real, word and short integers
  - *FADD, FSUB* - addition, subtraction
  - *FMUL, FDIV* - multiplication, division
  - *FSQRT* - square root
  - *FUCOM* - compare
  - *FCHS* - change sign
  - *FABS* - absolute value
  - *FSIN, FCOS* - sine, cosine
  - *FTAN, FATAN* - tangent, arcs tangent

## CPU FEATURES

- 100% software compatible with 80390 & 8051 industry standards
  - *LARGE mode* – 8051 instruction set
  - *FLAT mode* – 80390 instruction set
- Pipelined RISC architecture enables to execute instructions 15.55 times faster compared to standard 8051
- Up to 14.632 VAX MIPS at 100 MHz
- 24 times faster multiplication
- 12 times faster addition
- 2 Data Pointers (DPTR) for faster memory blocks copying
  - *Advanced INC & DEC modes*
  - *Auto-switch of current DPTR*
- Up to 256 bytes of internal (on-chip) Data Memory
- Up to 8M bytes of linear Program Memory
  - *64 kB of internal (on-chip) Program Memory*
  - *8 MB external (off-chip) Program Memory*
- Up to 16M bytes of external (off-chip) Data Memory
  - *Synchronous eXternal Data Memory (SXDM) Interface*
- User programmable Program Memory Wait States solution for wide range of memories speed
- User programmable External Data Memory Wait States solution for wide range of memories speed
- De-multiplexed Address/Data bus to allow easy connection to memory
- Dedicated signal for Program Memory writes.
- Interface for additional Special Function Registers
- Fully synthesizable, static synchronous design with positive edge clocking and no internal tri-states
- Scan test ready
- 2.0 GHz virtual clock frequency in a 0.25u technological process

## DELIVERABLES

- ◆ Source code:
  - VHDL Source Code or/and
  - VERILOG Source Code or/and
  - Encrypted, or plain text EDIF
- ◆ VHDL & VERILOG test bench environment
  - Active-HDL automatic simulation macros
  - ModelSim automatic simulation macros
  - Tests with reference responses
- ◆ Technical documentation
  - Installation notes
  - HDL core specification
  - Datasheet
- ◆ Synthesis scripts

- ◆ Example application
- ◆ Technical support
  - IP Core implementation support
  - 3 months maintenance
    - Delivery of the IP Core and documentation updates, minor and major versions changes
    - Phone & email support

## DESIGN FEATURES

### ◆ PROGRAM MEMORY:

The DP80390 soft core is dedicated to operate with Internal and External Program Memory. Its maximum linear size is equal to 8MB. Internal Program Memory can be implemented as:

- *ROM located in address range between  $0x0000 \div (ROM_{size}-1)$*
- *RAM located in address range between  $(64kB-RAM_{size}) \div 0xFFFF$*

External Program Memory can be implemented as ROM or RAM, located in address range between  $ROM_{size} \div 8$  MB excluding area occupied by  $RAM_{size}$ .

### ◆ INTERNAL DATA MEMORY:

The DP80390XP can address Internal Data Memory of up to 256 bytes The Internal Data Memory can be implemented as Single-Port synchronous RAM.

### ◆ EXTERNAL DATA MEMORY:

The DP80390XP soft core can address up to 16 MB of External Data Memory. Extra DPX (*Data Pointer eXtended*) register is used for segments swapping.

### ◆ USER SPECIAL FUNCTION REGISTERS:

Up to 60 External (user) Special Function Registers (ESFRs) may be added to the DP80390XP design. ESFRs are memory mapped into Direct Memory, between addresses 80 hex and FF hex in the same manner, as core SFRs and may occupy any address that is not occupied by a core SFR.

### ◆ WAIT STATES SUPPORT:

The DP80390XP soft core is dedicated for operation with wide range of Program and Data memories. Slow Program and External Data memory may assert a memory Wait signal to hold up CPU activity.

## LICENSING

Comprehensible and clearly defined licensing methods **without royalty-per-chip fees** make use of our IP Cores easy and simple.

*Single-Site license option* – dedicated to small and middle sized companies, which run their business in one place.

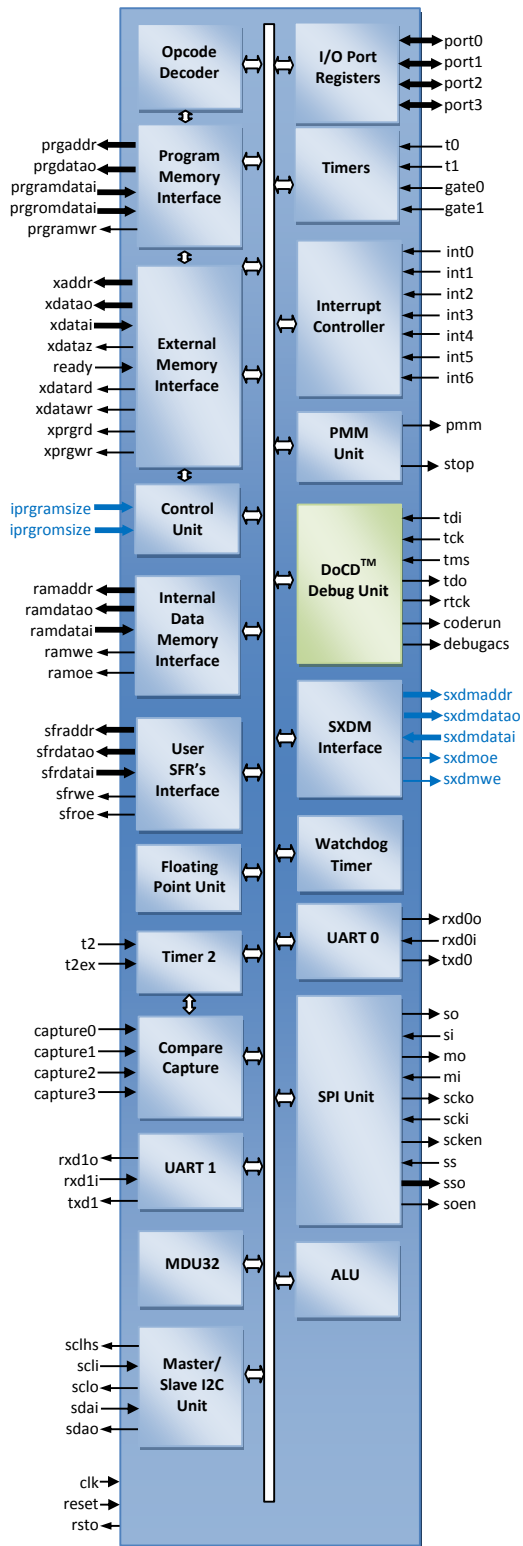
*Multi-Site license option* – dedicated to corporate customers, who operate at several locations. The licensed product can be used in selected company branches.

In all cases the number of IP Core instantiations within a project and the number of manufactured chips are unlimited. The license is royalty-per-chip free. There are no restrictions regarding the time of use.

There are two formats of the delivered IP Core:

- VHDL or Verilog RTL synthesizable source code called HDL Source code
- FPGA EDIF/NGO/NGD/QXP/VQM called Netlist

## BLOCK DIAGRAM



## CONFIGURATION

The following parameters of the DP80390XP core can be easily adjusted to requirements of a dedicated application and technology. Configuration of the core can be effortlessly done, by changing appropriate constants in the package file. There is no need to change any parts of the code.

- Internal Program Memory type
  - synchronous
  - asynchronous
- Internal Program ROM Memory size
  - 0 - 64kB
- Internal Program RAM Memory size
  - 0 - 64kB
- Internal Program Memory fixed size
  - true
  - false
- Second Data Pointer (DPTR1)
  - used
  - unused
- DPTR0 decrement
  - used
  - unused
- DPTR1 decrement
  - used
  - unused
- Data Pointers auto-switch
  - used
  - unused
- **Interrupts**
  - subroutines location
  - used
  - unused
- Timing access protection
  - used
  - unused
- Power Management Mode
  - used
  - unused
- Stop mode
  - used
  - unused
- DoCD™ debug unit
  - used
  - unused

Besides mentioned parameters, all available peripherals and external interrupts can be excluded from the core, by changing appropriate constants in the package file.

## PINS DESCRIPTION

PIN	TYPE	DESCRIPTION
clk	input	Global clock
reset	input	Global reset
port0i	input	Port 0 input
port1i	input	Port 1 input
port2i	input	Port 2 input
port3i	input	Port 3 input
ipgramsize	input	Size of on-chip RAM CODE
iprgromsize	input	Size of on-chip ROM CODE
prgramdata	input	Data bus from int. RAM prog. memory
prgromdata	input	Data bus from int. ROM prog. memory
sxdmdatai	input	Data bus from sync external data memory (SXDM)
xdatai	input	Data bus from external memories
ready	input	External memory data ready
ramdatai	input	Data bus from internal data memory
sfrdatai	input	Data bus from user SFR's
int0	input	External interrupt 0
int1	input	External interrupt 1
int2	input	External interrupt 2
int3	input	External interrupt 3
int4	input	External interrupt 4
int5	input	External interrupt 5
int6	input	External interrupt 6
t0	input	Timer 0 input
t1	input	Timer 1 input
t2	input	Timer 2 input
gate0	input	Timer 0 gate input
gate1	input	Timer 1 gate input
t2ex	input	Timer 2 gate input
capture0	input	Timer 2 capture 0 line
capture1	input	Timer 2 capture 1 line
capture2	input	Timer 2 capture 2 line
capture3	input	Timer 2 capture 3 line
rxdi0	input	Serial receiver input 0
rxdi1	input	Serial receiver input 1
scli	input	Master/Slave I2C clock line input
sdai	input	Master/Slave I2C data input
ss	input	SPI slave select
si	input	SPI slave input
mi	input	SPI master input
scki	input	SPI clock input
tdi	input	DoCD™ TAP data input
tck	input	DoCD™ TAP clock input
tms	input	DoCD™ TAP mode select input



<b>rsto</b>	output	Reset output
<b>port0o</b>	output	Port 0 output
<b>port1o</b>	output	Port 1 output
<b>port2o</b>	output	Port 2 output
<b>port3o</b>	output	Port 3 output
<b>prgaddr</b>	output	Internal program memory address bus
<b>prgdatao</b>	output	Data bus for internal program memory
<b>prgramwr</b>	output	Internal program memory write
<b>sxdmaddr</b>	output	Sync XDATA memory address bus (SXDM)
<b>sxdmdatao</b>	output	Data bus for Sync XDATA memory (SXDM)
<b>sxdmoe</b>	output	Sync XDATA memory read (SXDM)
<b>sxdmwe</b>	output	Sync XDATA memory write (SXDM)
<b>xaddr</b>	output	Address bus for external memories
<b>xdatao</b>	output	Data bus for external memories
<b>xdataz</b>	output	Turn xdata bus into 'Z' state
<b>xprgrd</b>	output	External program memory read
<b>xprgwr</b>	output	External program memory write
<b>xdatard</b>	output	External data memory read
<b>xdatawr</b>	output	External data memory write
<b>ramaddr</b>	output	Internal Data Memory address bus
<b>ramdatao</b>	output	Data bus for internal data memory
<b>ramoe</b>	output	Internal data memory output enable
<b>ramwe</b>	output	Internal data memory write enable
<b>sfraddr</b>	output	Address bus for user SFR's
<b>sfrdatao</b>	output	Data bus for user SFR's
<b>sfroe</b>	output	User SFR's read enable
<b>sfrwe</b>	output	User SFR's write enable
<b>tdo</b>	output	DoCD™ TAP data output
<b>rtck</b>	output	DoCD™ return clock line
<b>debugacs</b>	output	DoCD™ accessing data
<b>coderrun</b>	output	CPU is executing an instruction
<b>pmm</b>	output	Power management mode indicator
<b>stop</b>	output	Stop mode indicator
<b>rxd0o</b>	output	Serial receiver output 0
<b>rxd1o</b>	output	Serial receiver output 1
<b>txd0</b>	output	Serial transmitter output 0
<b>txd1</b>	output	Serial transmitter output 1
<b>sclo</b>	output	Master/Slave I2C clock output
<b>schs</b>	output	High speed Master I2C clock line
<b>sdao</b>	output	Master/Slave I2C data output
<b>sso[7:0]</b>	output	SPI slave select lines
<b>so</b>	output	SPI slave output
<b>mo</b>	output	SPI master output
<b>scko</b>	output	SPI clock output
<b>scken</b>	output	SPI clock line tri-state buffer control
<b>soen</b>	output	SPI slave output enable

## UNITS SUMMARY

**ALU** – Arithmetic Logic Unit - performs the arithmetic and logic operations, during execution of an instruction. It contains accumulator (ACC), Program Status Word (PSW), (B) registers and related logic, like arithmetic unit, logic unit, multiplier and divider.

**Opcode Decoder** – Performs an opcode decoding instruction and control functions for all other blocks

**Control Unit** – It performs the core synchronization and data flow control. This module is directly connected to Opcode Decoder and it manages execution of all microcontroller tasks.

**Program Memory Interface** – Program Memory Interface contains Program Counter (PC) and related logic. It performs the instructions code fetching. Program Memory can be also written. This feature allows usage of a small boot loader, to load new program into ROM, RAM, EPROM or FLASH EEPROM storage via UART, SPI, I2C or DoCD™ module.

**External Memory Interface** - Contains memory access related registers, such as Data Page High (DPH), Data Page Low (DPL)

and Data Page Pointer (DPP) registers. It performs the external Program and Data Memory addressing and data transfers. Program fetch cycle length can be programmed by the user. This feature is called Program Memory Wait States and it allows core, to work with different speed program memories.

**Synchronous eXternal Data Memory (SXDM) Interface** – contains XDATA memory access related logic, allowing fast access to synchronous memory devices. It performs the external Data Memory addressing and data transfers. This memory can be used to store large variables, frequently accessed by CPU, improving overall performance of application.

**Internal Data Memory Interface** – Interface controls access into the internal memory of size up to 256 bytes. It contains 8-bit Stack Pointer (SP) register and related logic.

**User SFRs Interface** – Special Function Registers interface controls access to the special registers. It contains standard and used defined registers and related logic. User defined external devices can be quickly accessed (read, written, modified), by using all direct addressing mode instructions.

**Interrupt Controller** – Interrupt Controller module is responsible for the interrupt manage system, for the external and internal interrupt sources. It contains interrupt related registers, such as Interrupt Enable (IE), Interrupt Priority (IP), Extended Interrupt Enable (EIE), Extended Interrupt priority (EIP) and (TCON) registers.

**I/O Ports** – Block contains 8051's general purpose I/O ports. Each of port's pin can be read/write as a single bit or as an 8-bit bus called P0, P1, P2, P3.

**Power Management Unit** – Power Management Unit contains advanced power saving mechanisms with switchback feature, allowing external clock control logic to stop clocking (Stop mode) or run core in lower clock frequency (Power Management Mode), to significantly reduce power consumption. Switchback feature allows UARTs and interrupts to be processed in full speed mode, if enabled. It is highly desirable, when microcontroller is planned to be used in portable and power critical applications.

**DoCD™ Debug Unit** – it's a **real-time hardware debugger**, which provides debugging capability of a whole SoC system. Unlike other on-chip debuggers, **DoCD™** ensures **non-intrusive debugging** of running application. It can halt, run, step into or skip an instruction, read/write any contents of microcontroller, including all registers, internal and external program memories and all SFRs, including user defined peripherals. Hardware breakpoints can be set and controlled on program memory, internal and external data memories, as well as on SFRs. Hardware watchpoints can be set and controlled on internal and external data memories and also on SFRs. Hardware watchpoints are executed, if any write/read occurs at particular address, with certain data pattern or without pattern. Two additional pins: CODERUN and DEBUGACS, indicate the state of the debugger and CPU. CODERUN is active, when CPU is executing an instruction. DEBUGACS pin is active, when any access is performed by **DoCD™** debugger. The **DoCD™** system includes **TTAG** or **JTAG interface** and complete set of tools, to communicate and work with core in real time debugging. It is built, as a scalable unit and some features can be turned off by the user, to save silicon and reduce power consumption. When debugger is not used, it is automatically switched to power save

mode. Finally, when debug option is no longer used, whole debugger is turned off.

**Floating Point Unit** – FPMU contains floating arithmetic point xIEEE-754, compliant instructions (C **float**, **int**, **long int** types supported). It is used to execute single precision floating point operations such as: addition, subtraction, multiplication, division, square root, comparison absolute value of number and change of sign. Basing on specialized CORDIC algorithm, full set of trigonometric operations is also allowed: sine, cosine, tangent, arctangent. It also has built-in integer to floating point and vice versa conversion instructions. FPU supports single precision real numbers, 16-bit and 32-bit signed integers. This unit has included standard software interface, which enables easy usage and interfacing with user's C/ASM written programs.

**MDU32 Multiply Divide Unit** – It is a fixed point, fast 16-bit and 32-bit multiplication and division unit. It supports unsigned and 2's complement signed integer operands. The MDU32 is controlled by dedicated direct memory access module (called DMA). All arguments and result registers are automatically read and written back by internal DMA. This unit has included standard software interface, which allows easy usage and interfacing with user C/ASM written programs. **This module is a modern replacement for older MDU.**

**Timers** – System timers module. Contains two 16bits configurable timers: Timer 0 (TH0, TL0), Timer 1 (TH1, TL1) and Timers Mode (TMOD) registers. In the timer mode, timer registers are incremented every 12 (or 4) CLK periods, when appropriate timer is enabled. In the counter mode, the timer registers are incremented every falling transition on their corresponding input pins (T0, T1), if gates are opened (GATE0, GATE1). T0, T1 input pins are sampled every CLK period. It can be used as clock source for UARTs.

**Timer 2** – Timer 2 - Second system timer module contains one 16-bit configurable timer: Timer 2 (TH2, TL2), capture registers (RLDH, RLDL) and Timer 2 Mode (T2MOD) register. It can work as a 16-bit timer / counter, 16-bit auto-reload timer / counter. It also supports compare capture unit (if present in the system). It can be used as clock source for UART0.

**Compare Capture Unit** – The compare/capture/reload unit is one of the most powerful peripheral units of the core. It can be used for all kinds of digital signal generation and event capturing, such as pulse generation, pulse width modulation, measurements etc.

**Watchdog Timer** – The watchdog timer is a 27-bit counter, which is incremented in every system clock period (CLK pin). It performs system protection against software upsets.

**UART0** – Universal Asynchronous Receiver and Transmitter module is full duplex, which means, it can transmit and receive concurrently. Includes Serial Configuration register (SCON), serial receiver and transmitter buffer (SBUF) registers. Its receiver is double-buffered, meaning, it can commence reception of a second byte, before the previously received byte has been read from the receive register. Writing to SBUF0 loads the transmit register and reading SBUF0, reads a physically separate receive register. Works in 3 asynchronous and 1 synchronous modes. UART0 can be synchronized by Timer 1 or Timer 2 (if present in the system).

**UART1** – Universal Asynchronous Receiver and Transmitter module is full duplex - it can transmit and receive concurrent-

ly. It includes Serial Configuration register (SCON1), serial receiver and transmitter buffer (SBUF1) registers. Its receiver is double-buffered, which means, it can commence reception of a second byte before the previously received byte has been read from the receive register. Writing to SBUF1, loads the transmit register and reading SBUF1, reads a physically separate receive register. Works in 3 asynchronous and 1 synchronous modes. UART1 is synchronized by Timer 1.

**Master I2C Unit** – I2C bus controller is a Master module. The core incorporates all features required by I2C specification. It supports both 7-bit and 10-bit addressing modes on the I2C bus and works as a master transmitter and receiver. It can be programmed to operate with arbitration and clock synchronization, to allow it to operate in multi-master systems. Built-in timer enables operation within wide range of the input frequencies. The timer allows achieving any non-standard clock frequency. The I2C controller supports all transmission modes: Standard, Fast, Fast+ and High Speed up to 3400kB/s.

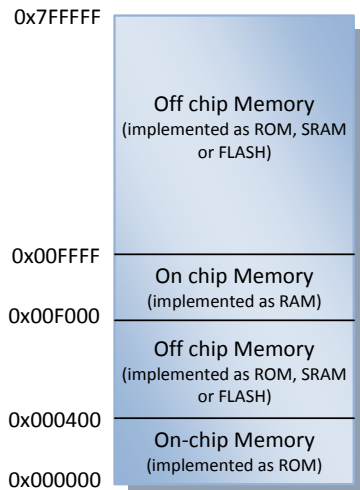
**Slave I2C Unit** – I2C bus controller is a Slave module. The core incorporates all features required by I2C specification. It works as a slave transmitter/receiver, depending on working mode, determined by a master device. The I2C controller supports all transmission modes: Standard, Fast, Fast+ and High Speed up to 3400kB/s.

**SPI Unit** – It is a fully configurable master/slave Serial Peripheral Interface, which allows user to configure polarity and phase of serial clock signal SCK. It allows the microcontroller to communicate with serial peripheral devices. It is also capable of interprocessor communication in a multi-master system. A serial clock line (SCK) synchronizes shifting and sampling of the information on the two independent serial data lines. SPI data are simultaneously transmitted and received. SPI system is flexible enough, to interface directly with numerous standard product peripherals, from several manufacturers. Data transfer rate up to CLK/4. Clock control logic allows to select the clock polarity and to choose the two fundamentally different clocking protocols, to accommodate most available synchronous serial peripheral devices. When the SPI is configured as a master, software selects one of four different bit rates for the serial clock. Error-detection logic is included, to support interprocessor communications. A write-collision detector indicates, when an attempt is made, to write data to the serial shift register, while the transfer is in progress. A multiple-master mode-fault detector automatically disables SPI output drivers, if more than one SPI device simultaneously attempts to become bus master.

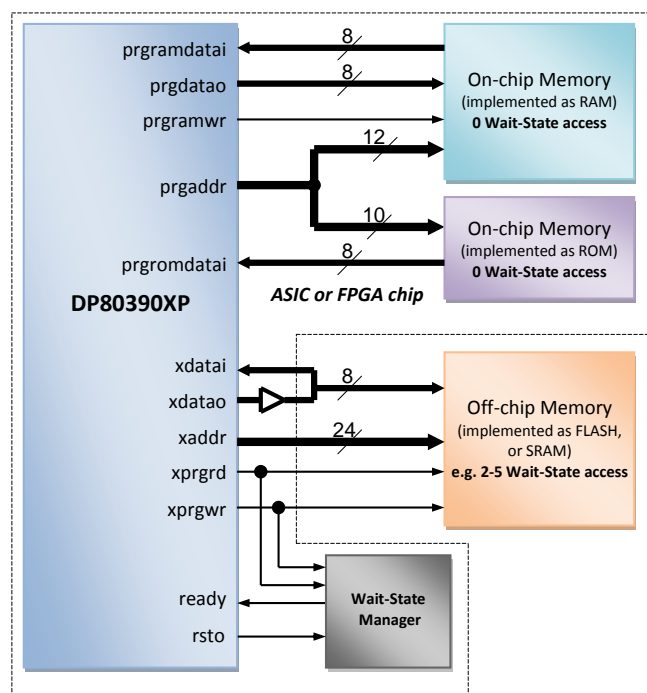
## PROGRAM CODE SPACE IMPLEMENTATION

The following figure shows an example Program Memory space implementation in systems with the DP80390XP Microcontroller core. The on-chip Program Memory located in the address space between 0kB and 1kB, is typically used for the BOOT code with system initialization functions. This part of the code is typically implemented as ROM. The on-chip Program Memory located in address space between 60kB and 64kB, is typically used for timing critical part of the code, e.g. interrupt subroutines, arithmetic functions etc. This part of the code is typically im-

plemented as RAM and can be loaded by the BOOT code during initialization phase, from an off-chip memory or through the RS232 interface from an external device. The program code is executed from the two spaces mentioned above without wait-states and can achieve top performance of up to 200 million instructions per second (many instructions executed in one clock cycle). The off-chip Program Memory located in the address space between 1kB and 60kB and above 64 kB, is typically used for the main code and constants. This part of the code is usually implemented as ROM, SRAM or a FLASH device. Due to relatively long access time, the program code executed from devices mentioned above must be fetched with additional Wait-States. The number of required Wait-States depends on memory access time and a DP80390XP clock frequency. In most cases, the proper number of Wait-States cycles is between 2-5. The READY pin can be also dynamically modulated e.g. by SDRAM controller.

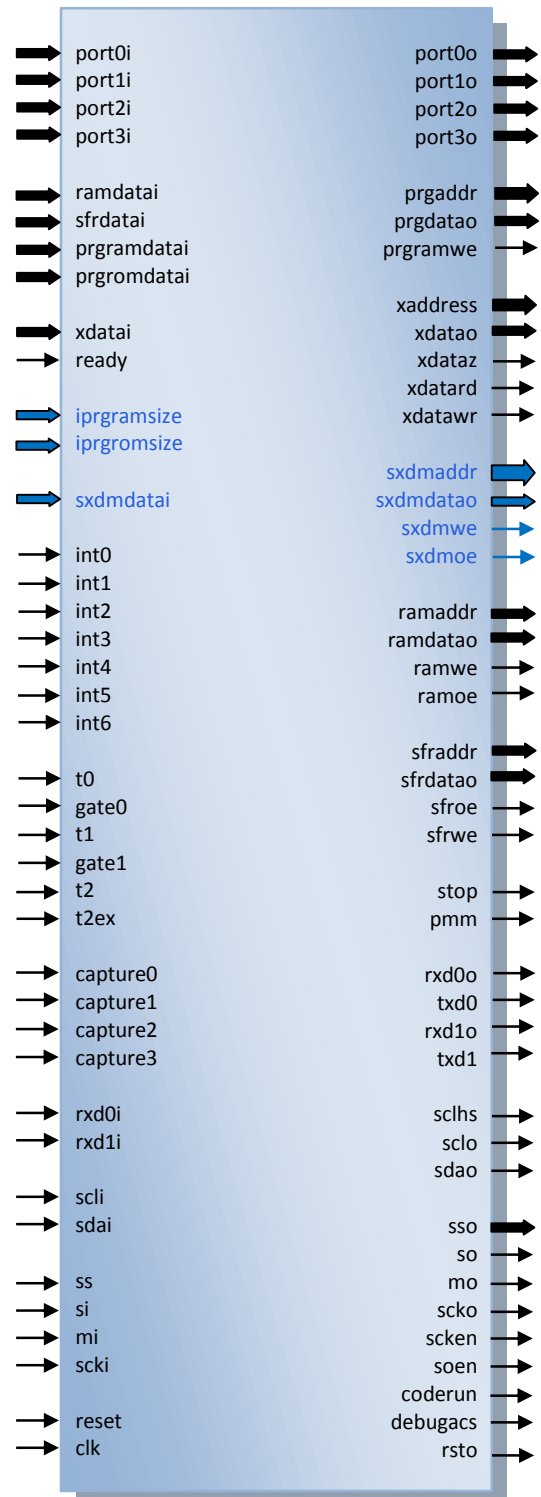


The figure below shows typical Program Memories connections in system with the DP80390XP Microcontroller core.



The implementation described above should be treated as an example only. All Program Memory spaces are fully configurable. For timing-critical applications, the whole program code can be implemented as on-chip ROM and (or) RAM and executed without Wait-States, but for some other applications, the entire program code can be implemented as off-chip ROM or FLASH and executed with required number Wait-State cycles.

## SYMBOL



## DP8051 FAMILY OVERVIEW

Main features of each DP80C51 family member have been summarized in the table below. It gives a brief member characteristic, helping you to select the most suitable IP Core for your application. You can specify your own peripheral set (including listed below and others) and requests the core modifications.

Design	Architecture speed grade	Program Memory space			Stack space size	Internal Data Memory space	External Data Memory space	External Data / Program Memory Wait States	Power Management Unit Interface for additional SFRs	Interrupt sources	Interrupt levels	Data Pointers	Timer/Counters	UART	\I/O Ports	Compare/Capture	Watchdog	Master I <sup>2</sup> C Bus Controller	Slave I <sup>2</sup> C Bus Controller	SPI	Fixed Point Coprocessor	Floating Point Coprocessor
		on-chip RAM	on-chip ROM	off-chip																		
DP8051CPU	10	64k	64k	64k	256	256	16M	✓	✓	✓	2	2	1	-	-	-	-	-	-	-	-	-
DP8051	10	64k	64k	64k	256	256	16M	✓	✓	✓	5	2	1	2	1	4	-	-	-	-	-	-
DP8051XP	10	64k	64k	64k	256	256	16M	✓	✓	✓	15	2	2	3	2	4	✓	✓	✓	✓	✓	✓

DP8051 family of Pipelined High Performance Microcontroller Cores

## DP80390 FAMILY OVERVIEW

Main features of each DP80390 family member have been summarized in the table below. It gives a brief member characteristic, helping you to select the most suitable IP Core for your application. You can specify your own peripheral set (including listed below and others) and requests the core modifications.

Design	Architecture speed grade	Program Memory space			Stack space size	Internal Data Memory space	External Data Memory space	External Data / Program Memory Wait States	Power Management Unit Interface for additional SFRs	Interrupt sources	Interrupt levels	Data Pointers	Timer/Counters	UART	\I/O Ports	Compare/Capture	Watchdog	Master I <sup>2</sup> C Bus Controller	Slave I <sup>2</sup> C Bus Controller	SPI	Fixed Point Coprocessor	Floating Point Coprocessor
		on-chip RAM	on-chip ROM	off-chip																		
DP80390CPU	10	64k	64k	8M	256	256	16M	✓	✓	✓	2	2	1	-	-	-	-	-	-	-	-	-
DP80390	10	64k	64k	8M	256	256	16M	✓	✓	✓	5	2	1	2	1	4	-	-	-	-	-	-
DP80390XP	10	64k	64k	8M	256	256	16M	✓	✓	✓	15	2	2	3	2	4	✓	✓	✓	✓	✓	✓

DP80390 family of Pipelined High Performance Microcontroller Cores

## PERFORMANCE

The following table gives survey about the Core area and performance in Programmable Logic Devices after Place & Route (CPU features and peripherals included):

Device	Speed grade	F <sub>max</sub>
ORCA 4E	-3	40 MHz

Core performance in LATTICE® devices

For the user, the most important factor is an application speed improvement. The most commonly used arithmetic functions and their improvement are shown in the table below. Improvement was computed as {80C51 clock periods} divided by {DP80390XP clock periods} required to execute an identical function. More details are available in core documentation.

Function	Improvement
8-bit addition (immediate data)	9,00
8-bit addition (direct addressing)	9,00
8-bit addition (indirect addressing)	9,00
8-bit addition (register addressing)	12,00
8-bit subtraction (immediate data)	9,00
8-bit subtraction (direct addressing)	9,00
8-bit subtraction (indirect addressing)	9,00
8-bit subtraction (register addressing)	12,00
8-bit multiplication	16,00
8-bit division	9,60
16-bit addition	12,00
16-bit subtraction	12,00
16-bit multiplication	13,60
32-bit addition	12,00
32-bit subtraction	12,00
32-bit multiplication	12,60
<b>Average speed improvement:</b>	<b>11,12</b>

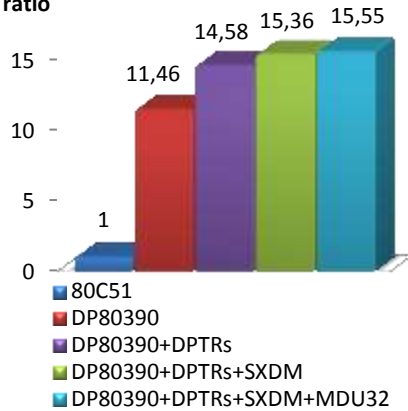
Dhrystone Benchmark Version 2.1 was used to measure the core performance. The following table shows the DP80390XP performance in terms of Dhrystone/sec and VAX MIPS rating.

Device	DMIPS/MHz	Ratio
80C51	0,00941	1,00
DP80390	0,10787	11,46
DP80390+DPTRs	0,13722	14,58
DP80390+DPTRs+SXDM	0,14457	15,36
DP80390+DPTRs+SXDM+MDU32	0,14632	15,55

Core performance in terms of Dhrystones



### VAX MIPS ratio



Area utilized by each unit of the DP80390XP core in vendor specific technologies is summarized in the table below.

Component	Area	
	[LC/PFU]	[FFs]
CPU*	1750	310
DPTR1 register	50	32
DPTR0 decrement	40	0
DPTR1 decrement	40	0
DPTR0 & DPTR1 auto-switch	30	8
Timed Access protection	20	10
Interrupt Controller	150	40
INT2-INT6	100	25
Power Management Unit	10	5
I/O ports	100	35
Timers	170	50
Timer 2	170	60
UART0	220	60
UART1	220	60
Master I2C Unit	270	120
Slave I2C Unit	170	70
SPI Unit	110	55
Compare Capture Unit	160	60
Watchdog Timer	100	45
Multiply Divide Unit 32	830	105
<b>Total area</b>	<b>4710/1300</b>	<b>1150</b>

#### Core components area utilization

\*CPU – consisted of ALU, Opcode Decoder, Control Unit, Program & Internal & External Memory Interfaces, User SFRs Interface

## CONTACT

### Digital Core Design Headquarters:

Wroclawska 94, 41-902 Bytom, POLAND

e-mail: [info@dcd.pl](mailto:info@dcd.pl)

tel.: 0048 32 282 82 66

fax: 0048 32 282 74 37

### Distributors:

Please check: <http://dcd.pl/sales>