Functional and Performance Validation of the 80S32 µC

80S32 Validation Report

Deliverable D4.1 - Report

Workpackage 4 – Validation

Author(s):	T.Lampaounas, C.Papadas, E.Politis, G.Dramitinos, K.Makris
Reviewer(s):	

Abstract:	The validation activity includes functional, usage and performance tests. Functional tests focus on verifying backward compatibility with the standard Intel 8032 core as well as correct operation of the additional features such as the CRC engine, external memory, interrupt management, timers and serial interfaces. Performance tests cover both distinct typical tasks involving matrix arithmetic, signal processing and sorting/searching algorithms among others as well as Dhrystone and Whetstone synthetic workloads.

Document Identifier:	D4.1/ISD/80S32	**Date of Delivery to ESA**	
Status:	Version 1.2	**Contractual:**	December 2008
Dissemination Level:	Restricted	**Actual:**	29 April 2009

Document history

Date	Author	Comments
28 February 2009	T. Lampaounas, E.Politis	Release to ESA
15 April 2009	T. Lampaounas, E.Politis	Added sections 2, 3 and 5.1
29 April 2009	T. Lampaounas, E.Politis, K.Makris	Added sections 4.2 & 6 and updated sections 4.3 & 6.

Table of Contents

1 Introduction

This document is the validation report for the 80S32 micro-controller (µC). It describes in detail the results obtained by the application of the tests described in the validation plan (document D2.3 v1.2.) The source code was developed and complied with the Keil µVision3 v3.60 toolset and is delivered in bundles of files (source code and configuration) referred to as "project" in Keil terminology.

1.1 Reference Documents

RD-1 "MCS51 MICROCONTROLLER FAMILY USER'S MANUAL", INTEL – (can be downloaded from http://developer.intel.com/design/mcs51/manuals/272383.htm)

RD-2 ADV80S32 Datasheet V2.5, ADV Engineering, 22.05.2001

RD-3 "80S32 Validation Plan", D2.3/ISD/80S32 V1.2, ISD SA, 15-02-2009

1.2 Glosary

Clk	Clock
CCSDS	Consultative Committee for Space Data System
CRC	Cyclic Redundancy Code
DC	Direct Current
DIP	Dual In-line Package
EDAC	Error Detection And Correction
FPGA	Field Programmable Gate Array
I/O	Input/Output
ISR	Interrupt Service Routine
Mhz	Megahertz. 106 hertz
MQFP	Metric Quad Flat Pack
NC	Not Connected/No Connection
NV	SRAM Non-Volatile Static Random Access Memory
PSW	Processor Status Word
PCB	Printed Circuit Board
PIO	Parallel Input Output
QFP	Quad Flat Pack
RAM	Random Access Memory
ROM	Read Only Memory
RS-232	One standard (protocol) for serial transmission
RxD	Received data
SFR	Special Function Register
SMD	Surface Mounted Device
SRAM	Static Random Access Memory
TC	Telecommand
TM	Telemetry

TxD	Transmitted data
µC	Micro-controller
USART	Universal Synchronous Asynchronous Receiver Transmitter
VCA	Virtual Channel Assembler
VCC	Voltage at the Common Collector. Positive operating voltage, connected to positive power supply.
VCCA	Vcc connection of the micro-controller's core voltage supply
VCCB	Vcc connection of the micro-controller's I/O voltage supply
VCCINT	Power supply for the internal core logic of the FPGA
VCCO	Power supply for the output drivers of the FPGA
VCM	Virtual Channel Multiplexer
VME	Verso Modulo Europa. Scalable backplane bus interface. Original specification IEEE 1014-1987.
VSS	Voltage for Substrate and Sources. Negative operating voltage, connected to negative power supply
VSSA	Ground connection of the micro-controller's core supply
VSSB	Ground connection of the micro-controller's I/O supply
D1.2	D1.2 delivery based to the Work Package 1 agreement

2 Toolset Configuration

The toolset features a device database holding configuration information for the tools. Selecting a device from the database, µVision sets all assembler, compiler, and linker options automatically. Information related to the memory map of the device has to be provided by the user.

The database has been extended with definitions for two devices, one supporting EDAC the other not, named 80S32-EDAC and 80S32 respectively. Their only difference is the amount of available internal memory, and XRAM.

The configuration consists of five fields, REGFILE, SFILE, CPU, MON and SIM though more can be added.

Field REGFILE specifies the header file defining the device’s register map and bit-fields.

Field SFILE specifies the default start-up file that should be included in a project. This file contains code and configuration information to initialize the memory, set the stack pointer and then transfer control to the main C function.

Field CPU defines the device characteristics, such as internal memory (IRAM), XRAM, clock frequency and the presence of double data pointers (MODP2). The MODP2 option enables the use of the second data pointer available in several Philips derivatives such as the NXP80C51RA+ device which has some similarities with 80S32.

Although the MODP2 option is meant to be used with Atmel and Philips devices such as the NXP-80C51RA+, it could also be used with 80S32 since both devices use the same mechanism to switch pointers. The DPTR switching register is mapped at different address, however this can be handled by redefining the DPSEL public symbol in the start-up file.

Fields MON and SIM are associated with the debugging features of the toolset. These are set to their default values, suitable for the Intel 8032AH device.

Name	80S32 - No EDAC
Description	8051 based CMOS controller with Dual DPTR, 32 I/O Lines, 3 Timers/Counters, 15 Interrupts/2 Priority Levels, ROM-Less, 256 Bytes on-chip RAM, 256 Bytes on-chip XRAM
Options	CPU=IRAM(0-0xFF) XRAM(0-0xFF) CLOCK(20000000) MODP2 MON=S8051.DLL TP51.DLL("-p52") REGFILE=80S32.H("ESA") SIM=S8051.DLL DP51.DLL("-p52") SFILE="LIB\80S32_STARTUP.A51" ("Standard 80S32 Startup Code")

Name	80S32 - With EDAC
Description	8051 based CMOS controller with Dual DPTR, 32 I/O Lines, 3 Timers/Counters, 15 Interrupts/2 Priority Levels, ROM-Less, 128 Bytes on-chip RAM, 128 Bytes on-chip XRAM
Options	CPU=IRAM(0-0x7F) XRAM(0-0x7F) CLOCK(20000000) MODP2 MON=S8051.DLL TP51.DLL("-p52") REGFILE=80S32.H("ESA") SIM=S8051.DLL DP51.DLL("-p52") SFILE="LIB\80S32_STARTUP.A51" ("Standard 80S32 Startup Code")

3 FPGA Configuration

The 80s32 validation board implements the interface between the uC and its peripheral devices. To this end a rich communication infrastructure has been designed, providing flexibility in uC connectivity.

The design implementation (FPGA) serves as a universal platform for the entire validation test suite (D2.3). However, there is a small number of tests with special needs requiring FPGA reconfiguration. The differences are minimal and they are implemented by enabling/disabling certain parts of the design. This is done by editing the source code and commenting-out or uncomment the appropriate lines.

Comments within the source code guide the user through the changes and provide instructions on how to treat the H/W in these special cases, such as test 1.0.3, USART test in a Loop back mode and debugging through a UART RS232 communication protocol.

4 Results

4.1 Software Tests

Test	Status	Files	Comments
Internal Memory			
1.1	Passed	IntMemory.zip	
1.2	Passed		
1.3	Problematic		see 5.2
1.4	Passed		
1.5	Passed	IntmemoryEDAC.zip	
External Memory			
2.1	Passed	ExMemory.zip	
Timers/Counters			
3.1	Passed	Timers.zip	
3.2	Passed		
3.3	Passed		
3.4	Passed		
3.5	Passed		
3.6	Passed		
3.7	Passed		
3.8	Passed		
3.9	Passed		
3.10	Passed		
3.11	Passed		
3.12	Passed		
3.13	Passed		
3.14	Passed		
3.15	Passed		
3.16	Passed		
3.17	Passed		
Serial Ports			
4.1	Passed	USART.zip	
4.2	Passed		
4.3	Passed		
4.4	Passed		
4.5	Passed		
4.6	Passed		
4.7	Passed		
4.8	Passed		
Interrupt Controller			
5.1	Problematic	Interrupt.zip	see 5.3

Test	Status	Files	Comments
CRC Accelerator Unit			
6.1	Passed	CRC.zip	
Instruction Set			
7.1 ~ 7.28	Passed	Instruction.zip	
Performance			
8.1	Passed	Performance.zip	
8.2	Passed		
8.3	Passed		
8.4	Passed		
8.5	Passed		
8.6	Passed		
8.7	Passed		
8.8	Passed		
8.9	Passed		
8.10	Passed		
8.11	Passed		
8.12	Passed		
8.13	Passed		
8.14	Passed		
8.15	Passed		
8.16	Passed		
8.17	Passed		
8.18	Passed		
8.19a	Passed		
8.19b	Passed		
8.20	Passed		
8.21	Passed		
8.22	Passed		
8.23	Passed		
8.24	Passed		
Electrical Power Consumption			
9.1	Passed	n/a	
9.2	Passed	SEUmemory.zip	
SEU			
10.1	Passed	SEU.zip	
10.2	Passed		

4.2 Performance Measurements

The test-suite includes tests specifically designed to produce performance figures for both distinct typical tasks and synthetic workloads. Typical tasks involve mathematical operations, matrix arithmetic, block data manipulation, sorting/searching algorithms, conditional branching, context switching, table lookup & interpolation, intensive pointer manipulation, encryption/decryption and signal processing algorithms. For all tests, the average execution

duration in machine cycles is returned. For synthetic workloads the popular Dhrystone & Whetstone benchmarks have been used.

Below are the results of the performance test with system clock set to 15MHz.

```
              P E R F O R M A N C E   T E S T
-------------------------------------------------------------
  (adjust TH0 reload value with respect to XTAL for T23 & T24)

Test 8.1 : 20 8bit arithmetic operations in 1660 cycles
Test 8.2 : 20 16bit arithmetic operations in 7174 cycles
Test 8.3 : 20 32bit arithmetic operations in 29448 cycles
Test 8.4 : 20 float arithmetic operations in 49271 cycles
Test 8.5 : 10 times a 3x4 to 4x5 matrix multip/ion in 325416 cycles
Test 8.6 : LU decomposition of a 5x5 matrix in 108996 cycles
Test 8.7 : Determinant of a 5x5 matrix in 140988 cycles
Test 8.8 : Copy of a 16x4 array of 8bit elements in 29061 cycles
Test 8.9 : Copy of a 16x4 array of 16bit elements in 23603 cycles
Test 8.10 : Copy of a 16x4 array of 32bit elements in 40178 cycles
Test 8.11 : BubbleSort    array[25]= 127692  array[100]= 371795 cycles
Test 8.12 : SelectionSort array[25]= 107681  array[100]= 313884 cycles
Test 8.13 : InsertionSort array[25]=  54420  array[100]= 139701 cycles
Test 8.14 : QuickSort     array[25]=  90308  array[100]= 182170 cycles
Test 8.15: 8bit branching 100 times in 5624 cycles
Test 8.16: 16bit branching 100 times in 14588 cycles
Test 8.17: 10 depth recursive function call 10 times in 2413 cycles
Test 8.18: Bilinear interpolation in 18347 cycles (average of 10)
Test 8.19a: Find Min/Max of 16bit 25 elem. D.Linked List in 249862 cycles
Test 8.19b: Search 100 val in 16bit 25 elem. D.Linked List in 266270 cycles
Test 8.20: FIR filter processing 150 16bit samples in 325273 cycles
Test 8.21: FFT (Radix-2 Cooley-Tukey) for 256 samples in 8398315 cycles
Test 8.22: iFFT (Radix-2 Cooley-Tukey) for 256 samples in 7677213 cycles
Test 8.23: 132 Dhrystones per second
Test 8.24: Single Whetstone 17 KIP's per second
```

4.3 Power Measurements

The performance test was used to obtain power and temperature measurements for different system clock frequencies. The power consumption of 80S32 was measured for different clock frequencies and various supply voltages, in two different states of activity. At the same time, the case temperature of DUT was constantly monitored and recorded in each test run. The maximum power consumption was determined using the 'Performance Test' suite as described in section 4.2.

The digital I/O pins of 80S32 are interfaced with the FPGA on the validation board via a set of level translators, for the proper adaptation of logic levels. The 'High' side of level translators (level shifters) between the FPGA and the uC is tied on the same power plane with that of uC. In the figures below, the consumption of the level shifters has been subtracted from the measured values and the figures only show the consumption of the 80S32 itself.

Definitions

- Idle state: FPGA is configured, uC receives clock but is held in reset condition (RESET* = 1)
- Running state: the uC executes the test routine 'Performance Test'.
- fCLK: clock frequency
- Vs: supply voltage (common between VCCA and VCCB pins of 80S32)
- Pi: 80S32 power consumption in idle state, as defined above
- Pr: 80S32 power consumption in running state, as defined above
- Tc: case temperature. The temperature measured on the case surface as described below (see Note 5).
- Tj: junction temperature

The following tables (Table 1, Table 2 and Table 3) illustrate the power consumption of 80S32 uC in the idle and running states, measured at spot clock frequencies and through the recommended voltage supply range (4.5 to 5.5V).

Vs = 4.5V				
f_{CLK} (MHz)	**P_i (mW)**	**P_r (mW)**	**Tc (ºC)** (see Notes 1,2)	**Tj (ºC)** (see Notes 1,6)
1	13,2	22,8	31,5	32,1
5	44,9	68,8	32,5	34,2
10	85,1	125,3	34	37,1
15	124,3	182,7	35,5	40
18	149,2	220	36,5	42
20	164,5	242,1	37	43,1
22	182,7	266	37,5	44,2

Table 1 - 80S32 power consumption for Vs=4.5V

Vs = 5.0V				
f_{CLK} (MHz)	**P_i (mW)**	**P_r (mW)**	**Tc (ºC)** (see Notes 1,2)	**Tj (ºC)** (see Notes 1,6)
1	17,9	28,5	31,5	32,2
5	58,3	88,1	33	35,2
10	108,3	160,5	35	39
15	160,5	233,9	37	42,8
18	200,9	280,7	38	45
20	210,5	309,4	38,5	46,2
22	231,7	339,2	39	47,5

Table 2 - 80S32 power consumption for Vs=5.0V

Vs = 5.5V				
f_{CLK} (MHz)	P_i (mW)	P_r (mW)	Tc (ºC) (see Notes 1,2)	Tj (ºC) (see Notes 1,6)
1	27,9	44,3	32	33,1
5	78,2	115,7	34	36,9
10	150,8	215,1	36	41,4
15	212,8	306,4	38,5	46,2
18	244,4	358	40	49
20	266,6	391,8	40,5	50,3
22	294,7	427	41,5	52,2

Table 3 - 80S32 power consumption for Vs=5.5V

Notes for the temperature measurements:

1. The temperature figures on Tables 5, 6 and 7 (Tc, Tj) are specified for the DUT in the running state.

2. The figures for Tc are stated using 0.5 ºC accuracy.

3. Ambient temperature (TA): the room temperature during measurements (measured with the same temperature probe as for the experiments) was 28ºC.

4. The temperature measurements were performed using a K-type thermocouple connected into a suitable digital read-out meter.

5. The temperature sensing location of the 80S32 DUT was the geometrical center of the ceramic package surface. This location was found to be the most representative spot for determining the case temperature of DUT, using the indicated temperature probing method. The thermocouple junction was properly fixed on the DUT socket in order to make good contact with the package surface.

6. Tj is calculated by assuming the thermal resistance of the 100-MQFP ceramic package to be 25ºC/W, since an exact value is not specified in [DR-2].

The 80S32 has reached a maximum Tc of 41.5ºC at 22MHz clock frequency and for 5.5V supply.

The power consumption of 80S32 is graphically represented in **Figure 1** and **Figure 2** for idle and running state respectively. The power consumption rate is about 16mW/MHz at the typical supply voltage of 5.0V for the running state.

The estimated power consumption of 80S32 at 10MHz is reported about 235mW [DR-4]. However, the corresponding supply voltage for this figure is not specified, neither the assumed activity state of the uC in which this theoretical figure was estimated. Table 4 summarizes the measured values at 10MHz within the specified supply range for comparison.

The estimated power figure for 80S32 consumption is more close to the actual figure measured at Vs = 5.5V, while the power consumption at the typical supply voltage (Vs = 5.0V) was measured lower than that. If this theoretical figure is specified for the typical supply voltage (Vs = 5.0V) we conclude that the actual power consumption of 80S32 was found to be lower than the power estimation made during the design phase.

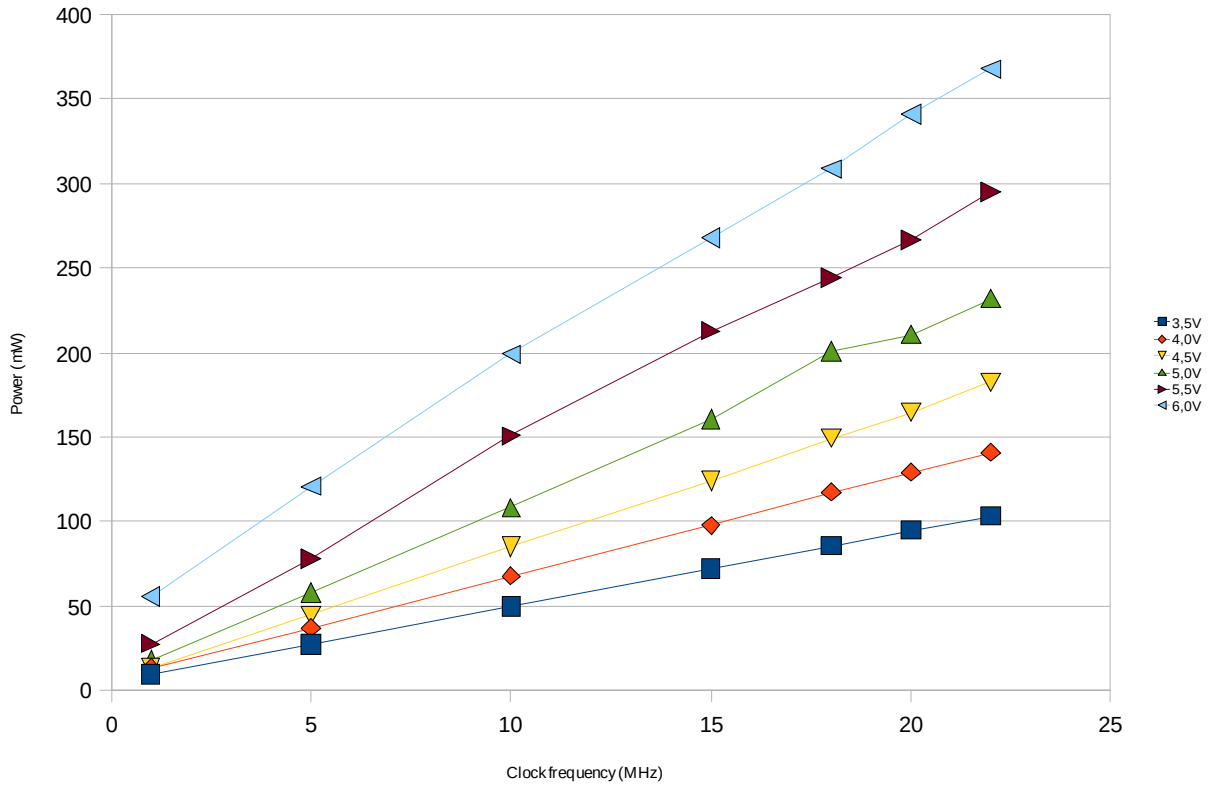

Figure 1 - 80S32 power consumption in idle state for various supply voltages.

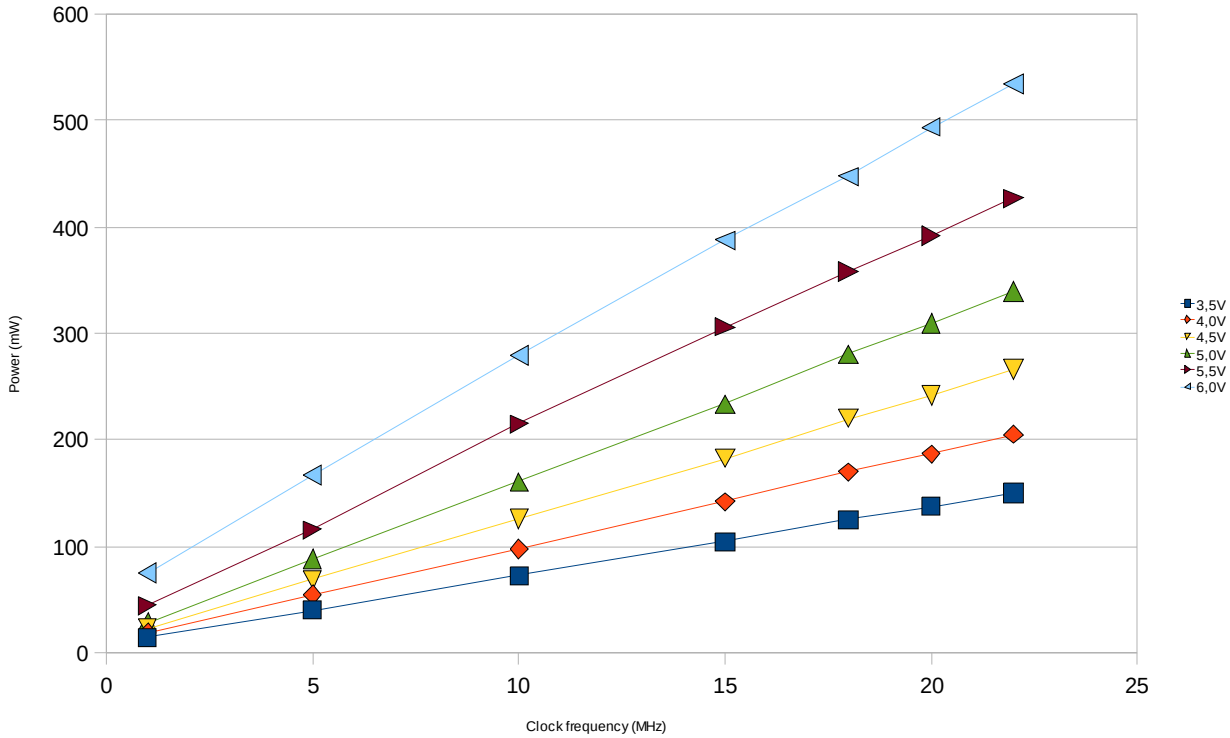

Figure 2 - 80S32 power consumption in running state for various supply voltages.

fCLK = 10MHz	
Vs (V)	**Pr (mW)**
4.5 (min)	125.3
5.0 (typ.)	160.5
5.5 (max)	215.1

Table 4 - Measured 80S32 power consumption at 10MHz

5 Comments

5.1 Difficulty to run compiled C code in EDAC mode

The SEU tests maintain a number of counters to store the number of detected errors as indicated by the EDAC hardware. The values of these counters are sent over the UART at regular intervals.

It has been observed that when the test was coded in C using the printf() library routine to communicate the counters, the displayed values were unexpected and obviously erroneous since the uC was not radiated at the time. This was confirmed by monitoring the interrupt activity, indicating no errors had taken place.

This behavior suggests memory corruption. Running the test on the emulator/debugger with “on-access” breakpoints set on the counters did not reveal any problem. Therefore, it is probable the corruption happens “indirectly” due to accessing the reserved memory area (EDAC) thus forcing the h/w to redirect the access to the user area where the counters are stored.

Given the simplicity of the test program it is likely that the problem stems from the pre-compiled libraries that are linked into the executable.

These problems have only been experienced when the EDAC for on-chip memory had been enabled. In all other cases, no such errors have been detected. The final version of the SEU test was implemented in assembly without using any libraries.

The failure to use C-compiled programs in EDAC mode is a serious issue and requires further investigation. As using breakpoints in the debugger did not provide the explanation, it is probably necessary to run a failing C-compiled program in VHDL or even netlist simulation so that any change to the memory locations can be detected.

5.2 Parallel Port P0

Parallel port P0 cannot be configured for general purpose I/O. According to the datasheet (see RD-2) setting register EXTBUS to 0x00 should reserve the port for I/O. Setting register P0DIR to 0xFF should configure the port as output, however no data are transmitted when the port is written. This has been observed through a logic analyzer by taping directly on the P0 pins. The port is also incapable to receive data (P0DIR set to 0x00) since it always holds value 0x00 although a different value is sent to the pins (also confirmed through the logic analyzer).

The port operates correctly in alternative function mode (address expansion) as it has demonstrated in test 2.1.

Ports P1, P2 & P3 have been found to operate correctly in both modes (see Timers & USART tests).

5.3 External Interrupts 2, 3 & 4

According to documentation (see RD-2) interrupts IE2, IE3 & IE4 can be triggered/cleared by software by setting appropriate bits in register TXCON1. However it has been found that once triggered the interrupt cannot be cleared (ISR is called over and over again) and instead it has to disabled through register IXEP3. This applies to all three external interrupts.

5.4 USART Interrupts

It has been observed through the use of a logic analyzer that the USART interrupt is raised (due to TI bit being set) as soon as the SXxBUFL register is written. Thus, when multiple bytes are to be sent, depending on the delay between consecutive writes some bytes will be overwritten and will not be transmitted. For example when 5 bytes were written (one at a time after interrupt had been raised), only the first and the last were actually transmitted. When a small delay was introduced between each write, then the third byte was also transmitted.

A similar problem exists when the USART is receiving, i.e. the interrupt is raised before the whole byte is received and the contents of SXxBUFL are 0x00.

5.5 Register Default Values

5.5.1 EXTBUS, P1CON, P2CON & P3CON

According to documentation (see RD-2) after power-up or reset, all parallel ports should be configured in alternative function mode, instead they are configured as bi-directional ports.

5.5.2 P0DIR, P1DIR, P2DIR & P3DIR

According to documentation (see RD-2) after power-up or reset, the direction of the parallel ports when in bi-directional mode should be output, instead they are configured for input.

5.5.3 TXCON2

According to documentation (see RD-2) if the register is written with value 0xAA, then we should get the same value when we read it. Instead we get value 0x00.

5.5.4 TXCON3

According to documentation (see RD-2) if the register is written with value 0xAA, then we should get the value 0x80 when we read it. Instead we get value 0x00.

5.6 Pin Assignment

5.6.1 CODE0_BUSCTL0, CODE1_BUSCTL1, SCANEN_BUSCTL2, FALLCLK_BUSCTL3

According to the documentation (see RD-2, Pin Assignment on pg.76) the following physical pins are defined as outputs which should be left floating:

- CODE0_BUSCTL0 (pin 97)
- CODE1_BUSCTL1 (pin 98)
- SCANEN_BUSCTL2 (pin 99)

- FALLCLK_BUSCTL3 (pin 100)

For proper operation these pins (especially the 'SCANEN_BUSCTL2' pin) should be connected directly to VSS, as discovered during the initial debugging phase of the design.

The first tests were performed by pulling all the above pins down via 10KΩ resistors, but glitches with duration shorter than 1 clock cycle have been observed on 'RD_n' & ' WR_n' lines. Inspection of the VHDL code and netlist revealed that the relevant pins are designed to be registered-out using a single scan-out multiplexer after the Flip-Flops. A voltage level of 0.5VDC was observed using laboratory measurements across the 10KΩ termination pull-down resistors. This voltage may appear due to the leakage current value of 50uA as specified in the Atmel datalog. This voltage level is considered to be close to the VIL threshold of the input pad. Additional noise could cause spurious activation of the pin and unwanted glitches as a consequence. The problem was addressed by short-circuiting the 10KΩ resistors. For normal operation these pins should be tied directly to the VSS plane of the PCB.

5.6.2 RESET

According to the documentation (see RD-2, Interface Signals on pg.71) the input pin 'RST' is erroneously specified as a reset suggesting active low triggering. The trigger mode of this pin is active high, as discovered during the initial debugging phase of the design.

6 Conclusions

The 80S32 µC extends the original 8052/51 device while maintaining code compatibility. Standard features such as, timers, I/O ports and UART, have been found to work properly, with the exception of port P0. In respect to the added features, there are problems with external interrupts IE2, IE3 & IE4 as well as the interrupts generated by the USART interfaces. Due to the latter, the program downloading capabilities of the µC (on-chip bootstrap) have not been tested.

The power consumption figures obtained by exercising the 80S32 within the specified voltage supply range were found to be comparable, but lower than the theoretical figure reported in [RD-4] during the design phase. The power consumption has a linear relationship with the clock frequency in both the specified (4.5 – 5.5V) and extended (3.5 – 6.0V) supply ranges, as expected.

Both the performance and power measurements tests were performed in different clock frequencies. The maximum clock frequency in which the 80S32 was able to operate normally was 22MHz, through the extended voltage supply range and is by 2MHz higher than the 20MHz figure reported in [RD-2].

A number of mistake with regards to pin description and register default values have been identified, requiring an update to the datasheet.

A few functional problems (external and USART interrupts) have been found requiring description in an errata section of the data sheet and definition of possible workarounds

The most serious issue is the difficulty to generate valid executables featuring EDAC support when complied from C-sources. This requires further investigation and definition of workaround, possibly requiring a patch to the development tools that would have to be defined in cooperation with the tool vendor.