u’nSP Instruction Set Summary

V1.1 - Nov 10, 2009

No.19, Industry E. Rd. IV, Hsinchu Science Park, Hsinchu City 30077, Taiwan, R.O.C.

Tel: +886-3-666-2118 Fax: +886-3-666-2117 Web: www.generalplus.com

Important Notice

GENERALPLUS TECHNOLOGY INC. reserves the right to change this documentation without prior notice. Information provided by GENERALPLUS TECHNOLOGY INC. is believed to be accurate and reliable. However, GENERALPLUS TECHNOLOGY INC. makes no warranty for any errors which may appear in this document. Contact GENERALPLUS TECHNOLOGY INC. to obtain the latest version of device specifications before placing your order. No responsibility is assumed by GENERALPLUS TECHNOLOGY INC. for any infringement of patent or other rights of third parties which may result from its use. In addition, GENERALPLUS products are not authorized for use as critical components in life support devices/ systems or aviation devices/systems, where a malfunction or failure of the product may reasonably be expected to result in significant injury to the user, without the express written approval of Generalplus.

Table of Content

Revision History

Revision	Date	By	Remark
1.0	2004/09/17		Original
1.1	2009/11/10	Nicole	Change Sunplus logo to Generalplus

1 Instruction Set Summary

Type	Syntax	ISA			F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
		1.1	1.2	2.0																
DSI6	DS=IM6				1	1	1	1	1	1	1	0	0	0	IM6					
CALL	CALL A22				1	1	1	1	-	-	0	0	0	1	A22[21:16]					
					A22[15:0]															
JMPF	GOTO A22				1	1	1	1	1	1	1	0	1	0	A22[21:16]					
					A22[15:0]															
JMPR	GOTO MR				1	1	1	1	1	1	1	0	1	1	-	-	-	-	-	-
FIR_MOV	FIR_MOV ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	1	0	$FIR_{(a)}$
Fraction	FRACTION ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	1	1	$FRA_{(b)}$
INT SET	INT FIQ/IRQ/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	0	$F_{(c)}$	$I_{(c)}$
IRQ	IRQ ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	0	0	$I_{(d)}$
SECBANK	SECBANK ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	0	1	$S_{(e)}$
FIQ	FIQ ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	1	$F_{(f)}$	0
IRQ Nest Mode	IRQNEST ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	1	$N_{(g)}$	1
BREAK	BREAK				1	1	1	1	-	-	-	1	0	1	1	-	-	0	0	0
CALLR	CALL MR				1	1	1	1	-	-	-	1	0	1	1	-	-	0	0	1
DIVS	DIVS MR,R2				1	1	1	1	-	-	-	1	0	1	1	-	-	0	1	0
DIVQ	DIVQ MR,R2				1	1	1	1	-	-	-	1	0	1	1	-	-	0	1	1
EXP	R2 = EXP R4				1	1	1	1	-	-	-	1	0	1	1	-	-	1	0	0
NOP	NOP				1	1	1	1	-	-	-	1	0	1	1	-	-	1	0	1
DS Access	DS=Rs/ Rs=DS				1	1	1	1	-	-	-	0	0	0	1	0	$W_{(h)}$	$Rs_{(i)}$		
FR Access	FR=Rs/ Rs=FR				1	1	1	1	-	-	-	0	0	0	1	1	$W_{(j)}$	$Rs_{(i)}$		
MUL	MR = Rd* Rs, {ss/us/uu}				1	1	1	$S_{Rs(k)}$	$Rd_{(i)}$			$S_{Rd(k)}$	0	0	0	0	1	$Rs_{(i)}$		
MULS	MR = [Rd]*[Rs], {ss/us/uu},N				1	1	1	$S_{Rs(k)}$	$Rd_{(i)}$			$S_{Rd(k)}$	1	$Size_{(l)}$				$Rs_{(i)}$		
Register BITOP	BITOP Rd,Rs				1	1	1	0	$Rd_{(i)}$			0	0	0	$Bitop_{(m)}$		0	$Rs_{(i)}$		
Register BITOP	BITOP Rd,offset				1	1	1	0	$Rd_{(i)}$			0	0	1	$Bitop_{(m)}$		Offset			
Memory BITOP	BITOP {D:}[Rd],offset				1	1	1	0	$Rd_{(i)}$			1	1	$D_{(n)}$	$Bitop_{(m)}$		Offset			
Memory BITOP	BITOP {D:}[Rd],Rs				1	1	1	0	$Rd_{(i)}$			1	0	$D_{(n)}$	$Bitop_{(m)}$		0	$Rs_{(i)}$		
Memory BITOP	BITOP {D:}[A16],offset				1	1	1	1	-	$D_{(n)}$	1	0	0	1	$Bitop_{(m)}$		Offset			
					A16															
16 bits Shift	Rd=Rd LSFT Rs				1	1	1	0	$Rd_{(i)}$			1	0	$LSFT_{(o)}$			1	$Rs_{(i)}$		
RETI	RETI				1	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0
RETF	RETF				1	0	0	1	1	0	1	0	1	0	0	1	0	0	0	0
Base+Disp6	Rd = Rd op [BP+IM6]				$OP_{(p)}$				$Rd_{(i)}$			0	0	0	IM6					
IMM6	Rd = Rd op IM6				$OP_{(p)}$				$Rd_{(i)}$			0	0	1	IM6					
Branch	Jxx label				$OP_{(q)}$				1	1	1	0	0	$D_{(r)}$	IM6					

Type	Syntax	ISA			F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
		1.1	1.2	2.0																
Stack Operation	Push/Pop R_H,R_L to [Rs]				$OP_{(s)}$				$R_H = Rd_{(i)}$			0	1	0	$Size_{(t)}$			$Rs_{(i)}$		
DS_Indirect	Rd = Rd op D:[Rs@]				$OP_{(p)}$				$Rd_{(i)}$			0	1	1	$D_{(n)}$	$@_{(u)}$		$Rs_{(i)}$		
IMM16	Rd = Rs op IMM16				$OP_{(v)}$				$Rd_{(i)}$			1	0	0	0	0	1	$Rs_{(i)}$		
					IMM16															
Direct16	Rd = Rs op [A16] (W = 0)				$OP_{(v)}$				$Rd_{(i)}$			1	0	0	0	1	W	$Rs_{(i)}$		
	[A16] = Rd op Rs (W = 1)				A16															
Direct6	Rd = Rd op [A6]				$OP_{(p)}$				$Rd_{(i)}$			1	1	1	A6					
Register	Rd = Rd op Rs SFT N				$OP_{(p)}$				$Rd_{(i)}$			1	$SFT_{(w)}$			$SFC_{(x)}$		$Rs_{(i)}$		
Ext Code(y)					1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
Ext Register	Ra=Ra op Rb				$OP_{(p)}$				$Ra_{2\text{-}0(i)}$			$Ra_{3(i)}$	0	0	0	0	$Rb_{(i)}$			
Ext Push/Pop	Push R_H, R_L to [Rb](W = 1)				W	$Size_{(t)}$			$R_H = Rx_{(i)}$			0	0	0	1	0	$Rb_{(i)}$			
	Pop R_H, R_L to [Rb](W = 0)																			
Ext IMM16	Ra=Rb op IMM16				$OP_{(v)}$				$Ra_{2\text{-}0(i)}$			$Ra_{3(i)}$	0	1	0	0	$Rb_{(i)}$			
					IMM16															
Ext A16	Ra=Rb op [A16] (W=0)				$OP_{(v)}$				$Ra_{2\text{-}0(i)}$			$Ra_{3(i)}$	0	1	1	W	$Rb_{(i)}$			
	[A16] = Ra op Rb(W=1)				A16															
Ext DS_Indirect	Rx=Rx op D:[Ry@]				$OP_{(p)}$				$Rx_{(i)}$			0	1	0	$D_{(n)}$	$@_{(u)}$		$Ry_{(i)}$		
Ext IM6	Rx=Rx op IM6				$OP_{(p)}$				$Rx_{(i)}$			1	1	0	IM6					
Ext Base+Disp6	Rx=Rx op [BP+IM6]				$OP_{(p)}$				$Rx_{(i)}$			0	1	1	A6					
Ext A6	Rx=Rx op [A6]				$OP_{(p)}$				$Rx_{(i)}$			1	1	1	A6					

(a.)

Syntax	FIR
FIR_MOV ON	0
FIR_MOV OFF	1

(b.)

Syntax	FRA
FRACTION OFF	0
FRACTION ON	1

(c.)

Syntax	FIQ	IRQ	F	I
INT OFF	OFF	OFF	0	0
INT IRQ	OFF	ON	0	1
INT FIQ	ON	OFF	1	0
INT FIQ,IRQ	ON	ON	1	1

(d.)

Syntax	I
IRQ OFF	0
IRQ ON	1

(e.)

Syntax	S
SECBANK OFF	0
SECBANK ON	1

(f.)

Syntax	F
FIQ OFF	0
FIQ ON	1

(g.)

Syntax	N
IRQNEST OFF	0
IRQNEST ON	1

(h.)

Syntax	W
Rs = DS	0
DS = Rs	1

(i.)

Syntax	Rs/Rd	Ra/Rb	Rx/Ry
SP	000	0000	
R1	001	0001	
R2	010	0010	
R3	011	0011	
R4	100	0100	
BP	101	0101	
SR	110	0110	
PC	111	0111	
R8		1000	000
R9		1001	001
R10		1010	010
R11		1011	011
R12		1100	100

Syntax	Rs/Rd	Ra/Rb	Rx/Ry
R13		1101	101
R14		1110	110
R15		1111	111

(j.)

Syntax	W
Rs = FR	0
FR = Rs	1

(k.)

Syntax	$\{S_{Rs},S_{Rd}\}$	Rd*Rs/[Rd]*[Rs]	ISA		
			1.1	1.2	2.0
uu	00	Unsigned*Unsigned	Not Support	Support	
us	10	Unsigned*Signed	Support		
ss	11	Signed*Signed			

(l.)

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Size	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	0000

(m.)

BITOP	00	01	10	11
Syntax	TSTB	SETB	CLRB	INVB

(n.)

Syntax	D	Description
D:[]	1	The MSB 6-bits of accessing address is data segment (DS).
[]	0	The MSB 6-bits of accessing address is zero.

(o.)

LSFT	000	001	010	011	100	101	110	111
Syntax	ASR	ASROR	LSL	LSLOR	LSR	LSROR	ROL	ROR

(p.)

A : Rd / Ra / Rx

B : [BP+IM6] / IM6 / {D:}[Rs@] / [A6] / Rs SFT N / Rb / {D:}[Ry@]

OP	Type	Syntax	N	Z	S	C
0000	ADD	A += B;	●	●	●	●
0001	ADC	A += B, Carry;	●	●	●	●
0010	SUB	A -= B;	●	●	●	●
0011	SBC	A -= B, Carry;	●	●	●	●
0100	CMP	CMP A, B;	●	●	●	●
0110	NEG	A =- B;	●	●	-	-
1000	XOR	A ^= B;	●	●	-	-
1001	LOAD	A = B;	●	●	-	-
1010	OR	A | = B;	●	●	-	-
1011	AND	A &= B;	●	●	-	-
1100	TEST	TEST A, B;	●	●	-	-
1101	STORE	B = A;	-	-	-	-

(q.)

OP	Syntax	Condition	Description
0000	JCC	C==0	carry clear
0000	JB	C==0	below (unsigned)
0000	JNAE	C==0	not above and equal (unsigned)
0001	JCS	C==1	carry set
0001	JNB	C==1	not below (unsigned)
0001	JAE	C==1	above and equal (unsigned)
0010	JSC	S==0	sign clear
0010	JGE	S==0	great and equal (signed)
0010	JNL	S==0	not less (signed)
0011	JSS	S==1	sign set
0011	JNGE	S==1	not great than (signed)
0011	JL	S==1	Less (signed)
0100	JNE	Z==0	not equal
0100	JNZ	Z==0	not zero
0101	JZ	Z==1	zero
0101	JE	Z==1	equal
0110	JPL	N==0	plus
0111	JMI	N==1	minus
1000	JBE	Not (Z==0 and C==1)	below and equal (unsigned)
1000	JNA	Not (Z==0 and C==1)	not above (unsigned)
1001	JNBE	Z==0 and C==1	not below and equal (unsigned)
1001	JA	Z==0 and C==1	above (unsigned)
1010	JLE	Not (Z==0 and S==0)	less and equal (signed)

OP	Syntax	Condition	Description
1010	JNG	Not (Z==0 and S==0)	not great (signed)
1011	JNLE	Z==0 and S==0	not less and equal (signed)
1011	JG	Z==0 and S==0	great (signed)
1100	JVC	N == S	not overflow (signed)
1101	JVS	N != S	overflow (signed)
1110	JMP	Always	unconditional branch
0000	JCC	C==0	carry clear
0000	JB	C==0	below (unsigned)

(r.)

D	Description
0	Forward jump
1	Backward jump

(s.)

OP	Syntax
1101	PUSH
1001	POP

(t.) Size : Register counts from RH to RL

(u.)

@	00	01	10	11
Syntax	Rs	Rs--	Rs++	++Rs

(v.)

Syntax	A	B	C
Rd = Rs op IMM16	Rd	Rs	IMM16
Rd = Rs op [A16]	Rd	Rs	[A16]
[A16] = Rd op Rs	[A16]	Rs	Rd
Ra = Rb op IMM16	Ra	Rb	IMM16
Ra = Rb op [A16]	Ra	Rb	[A16]
[A16] = Ra op Rb	[A16]	Rb	Ra

OP	Type	Syntax	N	Z	S	C
0000	ADD	A = B + C;	•	•	•	•
0001	ADC	A = B + C, Carry;	•	•	•	•
0010	SUB	A = B−C;	•	•	•	•
0011	SBC	A = B−C, Carry;	•	•	•	•
0100	CMP	CMP B, C;	•	•	•	•
0110	NEG	A =− C;	•	•	-	-
1000	XOR	A = B ^ C;	•	•	-	-
1001	LOAD	A = C;	•	•	-	-
1010	OR	A = B | C;	•	•	-	-
1011	AND	A = B & C;	•	•	-	-
1100	TEST	TEST B, C;	•	•	-	-
1101	STORE	C = B;	-	-	-	-

(w.)

Shift with Barrel shifter **:**

SFT	000	001	010	011	100	101
Syntax	NOP	ASR	LSL	LSR	ROL	ROR

(x.)

SFC	00	01	10	11
N	1	2	3	4

(y.)

Use 0xFF80 as extension prefix code and followed with extend instruction.

2 Comparison BETWEEN u'nSP VERSIONS

	ISA 1.0	ISA 1.1	ISA 1.2	ISA 2.0
Memory Bus	Single Bus	Single Bus	Single Bus	Separate Inst/Data
Address depth	22 bits	22 bits	22 bits	22 bits / 22bits
Data width	16 bits	16 bits	16 bits	16 bits / 16 bits
Pipeline	No	No	No	4 stage pipeline
General Registers(R1-R4)	Yes	Yes	Yes	Yes
System Registers (SP, BP, SR, PC)	Yes	Yes	Yes	Yes
Second Bank Registers (SR1-SR4)	No	No	Yes	Yes
Inner Flag Register (FR)	No	No	Yes	Yes
Extend Registers (R8-R15)	No	No	No	Yes
Interrupt Sources	10 (FIQ,IRQ,BRK)	10 (FIQ,IRQ,BRK)	10 (FIQ,IRQ,BRK)	10 (FIQ,IRQ,BRK)
Nested IRQ	No	No	Yes	Yes
Average CPI	6	5	5	2
MAC operation	signed x signed signed x unsigned	signed x signed signed x unsigned	signed x signed signed x unsigned unsigned x unsigned	signed x signed signed x unsigned unsigned x unsigned
MAC Cycles	13	12	12/13(uxu)	1
Guard bits	No	No	4	4
Fraction mode	No	No	Yes	Yes
Division	No	No	1- bit Non-restoring division DIVS : 2 cycles DIVQ : 3 cycles	1 bit Non-restoring division DIVS : 1 cycle DIVQ : 1cycle
EXP	No	No	Yes	Yes
Bit operation	No	No	Register / Memory	Register / Memory
16 bits shifter	No	No	Yes	Log Shifter
DS access	No	No	Yes	Yes
FR access	No	No	Yes	Yes
Far jump	Yes	Yes	Yes	Yes
Far indirect jump	No	No	Yes	Yes
Far indirect call	No	No	Yes	Yes
Extend Operations	No	No	No	Yes
Immediate (I6 / I16)	Yes	Yes	Yes	Yes
Direct (A6 / A16)	Yes	Yes	Yes	Yes
Indirect (DS indirect)	Yes	Yes	Yes	Yes
Relative (BP+IM6)	Yes	Yes	Yes	Yes
Multiple indirect (Push/Pop)	Yes	Yes	Yes	Yes