

SpixTools

User's Manual

v1.0.1

Copyright 2006 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303, U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copy-

Commands

sdas (1sh)

NAME

sdas

Commands

Commands

spixstats (1sh)

NAME

spixstats -

NAME

spix_sparc_dis, spix_sparc_dis32 – Disassemble a SPARC instruction

SYNOPSIS

```
#include <spix_sparc.h>
```

```
size_t spix_sparc_dis(char *buf, size_t szbuf, spix_sparc_iop_t iop, const void *pinst,  
                    spix_addr64_t addr);
```

```
size_t spix_sparc_dis32(char *buf, size_t szbuf, spix_sparc_iop_t iop, const void *pinst,  
                       spix_addr32_t addr);
```

DESCRIPTION

These functions disassemble a single SPARC instruction into the given buffer. The buffer *buf* must have size *szbuf*. The instruction must be represented by the opcode value *iop*, which is the value returned by a previous call to **spix_sparc_iop(3sh)**. The pointer *pinst* must point to the beginning of the instruction, and *addr* must be the instruction's virtual address.

Both functions write the disassembly to *buf*. If there is not enough room in *buf*, only the first *szbuf*

NAME

spix_sparc_iop – Return a SPARC instruction's opcode value

SYNOPSIS

```
#include <spix_sparc.h>
```

```
spix_sparc_iop_t spix_sparc_iop(spix_sparc_ver_t ver, const void *pinst);
```

DESCRIPTION

The **spix_sparc_iop()** function calculates the an "opcode" value for a SPARC instruction. This opcode

Spix Library

spix_sparc_iop_istype(3sh)

NAME

spix_sparc_iop_istype

This type selects all instructions that are defined for the SPARC V9 architecture. (It does not include any of the UltraSPARC extended instructions.)

SPIX_SPARC_ITYPE_VIS

This type selects all the UltraSPARC extended "vis" instructions.

SPIX_SPARC_ITYPE_PRIV

This type selects all of the privileged instructions. It does not select instructions that are privileged only for certain operands (such as RDASR) or that are only privileged depending on the processor's configuration (such as RTICK).

SPIX_SPARC_ITYPE_BAA

This type selects all branch instructions that are contingent on the value of a register.

SPIX_SPARC_ITYPE_FPOP1

This type selects all FPOP1 instructions as defined by the SPARC Architecture Manual.

SPIX_SPARC_ITYPE_ALU

This type selects all instructions that perform an integer arithmetic or logical operation. This does not include load, store, branch, or floating point instructions.

SPIX_SPARC_ITYPE_ILOAD

This type selects all instructions that load a value from memory into an integer register. This includes all atomic load/store instructions.

SPIX_SPARC_ITYPE_ISTORE

This type selects all instructions that conditionally or unconditionally store a value from an integer register into memory. This includes all atomic load/store instructions.

Note, the Shade instruction class **SHADE_ICLASS_IWSTART** does not have a corresponding instruc-

SEE ALSO

spix_sparc_iop(3sh), shade_iset(3sh).

Spix Library

spix_sparc_iop_memsize(3sh)

NAME

iop

sparc_iop_3sh)

NAME

spix_sparc_iop_name, spix_sparc_iop_Lname – Return the name for a SPARC instruction opcode

SYNOPSIS

```
#include <spix_sparc.h>
```

```
const char *spix_sparc_iop_name(spix_sparc_iop_t iop);
```

```
extern const size_t spix_sparc_iop_Lname;
```

DESCRIPTION

The `spix_sparc_iop_name()`

NAME

spix_sparc_iop_regpos – Return register usage au4dCion in SPARC instrucCion)

SPIX_SPARC_RUPOS_RD

Register is in the RD position.

SPIX_SPARC_RUPOS_IMP

The instruction does not explicitly reference the register, but its use is implied.

The third parameter to the *pfun* function is the register number. The function must interpret this register as either an integer, floating point, or special register number according to the value of the first parameter. If this is special register, the third parameter has one of the following values.

SPIX_SPARC_SREG_Y

The %y register.

SPIX_SPARC_SREG_ASI

The %asi register.

SPIX_SPARC_SREG_CCR

The %ccr register.

SPIX_SPARC_SREG_FPRS

The %fprs register.

SPIX_SPARC_SREG_FCC

One of the condition code fields of the %fsr register.

SPIX_SPARC_SREG_RM

The rounding mode field of the %fsr.

SPIX_SPARC_SREG_FSR

Another other field of the %fsr.

SPIX_SPARC_SREG_TICK

The %tick register.

SPIX_SPARC_SREG_GSR

The UltraSPARC extended %gsr register.

The final parameter to the *pfun* function is the *pdata* value passed to **spix_sparc_reguse()**.

sparc
_

NAME

spixcounts – spixcounts file format

SYNOPSIS

```
#include <spixcounts.h>
```

DESCRIPTION

The **spixcounts** file format is generated by the **spixcounts(1sh)** Shade analyzer, and is consumed by several of the SpixTools. This file format contains three parts: a header, basic block lengths, and basic

SEE ALSO

spixcounts(1sh), sadd(1sh), sdas(1sh), spixstats(1sh), sprint(1sh).