2.4 Status Register ST1

The following figure shows the bit fields of status register ST1. All of these bit fields are modified in the decode 2 phase of the pipeline. Detailed descriptions of these bits follow the figure.

Table 2-12. Bit Fields of Status Register 1 (ST1)

15		13	12	11	10	9	8
ARP			XF	MOM1MAP	Reserved	OBJMODE	AMODE
R/W-000			R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
IDLESTAT	EALLOW	LOOP	SPA	VMAP	PAGE0	DBGM	INTM
R-0	R/W-0	R-0	R/W-0	R/W-1	R/W-0	R/W-1	R/W-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

ARP (Bits 15-13) — Auxiliary register pointer.

This 3-bit field points to the current auxiliary register. This is one of the 32-bit auxiliary registers (XAR0-XAR7). The mapping of ARP values to auxiliary registers is as follows:

000 = XAR0 (selected at reset)

001 = XAR1

010 = XAR2

011 = XAR3

100 = XAR4

101 = XAR5

110 = XAR6

111 = XAR7

XF (Bit 12) —XF status bit.

This bit reflects the current state of the XFS output signal, which is compatible to the C2XLP CPU. This bit is set by the "M SETC XF"M instruction. This bit is cleared by the "M CLRC XF"M instruction. The pipeline is not flushed when setting or clearing this bit using the given instructions. This bit can be saved and restored by interrupts and when restoring the ST1 register. This bit is set to 0 on reset.

NOTE: Use of the XFS signal requires an external pin that is only present on TMS320x2801x devices.

M0M1MAP (Bit 11) —M0 and M1 mapping mode bit.

The M0M1MAP bit should always remain set to 1 in the C28x object mode. This is the default value at reset. The M0M1MAP bit may be set low when operating in C27x-compatible mode. The effect of this bit, when low, is to swap the location of blocks M0 and M1 only in program space and to set the stack pointer default reset value to 0x000. C28x mode users should never set this bit to 0.

Reserved (Bit 10) —Reserved.

This bit is reserved. Writes to this bit have no effect.

OBJMODE (Bit 9) — Object compatibility mode bit.

This mode is used to select between C27x object mode (OBJMODE == 0) and C28x object mode (OBJMODE == 1) compatibility. This bit is set by the "M C28OBJ"M (or "M SETC OBJMODE"M) instructions. This bit is cleared by the "M C27OBJ"M (or "M CLRC OBJMODE"M) instructions. The pipeline is flushed when setting or clearing this bit using the given instructions. This bit is saved and restored by interrupts and when restoring the ST1 register. This bit is set to 0 on reset.

AMODE (Bit 8) — Address mode bit.

This mode, in conjunction with the PAGE0 mode bit, is used to select the appropriate addressing mode decodes. This bit is set by the "LPADDR"M ("M SETC AMODE"M) instructions. This bit is cleared by the "M C28ADDR"M (or "M CLRC AMODE"M) instructions. The pipeline is not flushed when setting or clearing this bit using the given instructions. This bit is saved and restored by interrupts and when restoring the ST1 register. This bit is set to 0 on reset.

Note: Setting PAGE0 = AMODE = 1 will generate an illegal instruction trap ONLY for instructions that decode a memory or register addressing mode field (loc16 or loc32).

IDLESTAT (Bit 7) — IDLE status bit.

This read-only bit is set when the IDLE instruction is executed. It is cleared by any one of the following events:

- An interrupt is serviced.
- An interrupt is not serviced but takes the CPU out of the IDLE state.
- A valid instruction enters the instruction register (the register that holds the instruction currently being decoded).
- A device reset occurs.

When the CPU services an interrupt, the current value of IDLESTAT is saved on the stack (when ST1 is saved on the stack), and then IDLESTAT is cleared. Upon return from the interrupt, IDLESTAT is not restored from the stack.

EALLOW (Bit 6) — Emulation access enable bit.

This bit, when set, enables access to emulation and other protected registers. Set this bit by using the EALLOW instruction and clear this bit by using the EDIS instruction. See the data sheet for a particular device to determine the registers that are protected.

When the CPU services an interrupt, the current value of EALLOW is saved on the stack (when ST1 is saved on the stack), and then EALLOW is cleared. Therefore, at the start of an interrupt service routine (ISR), access to protected registers is disabled. If the ISR must access protected registers, it must include an EALLOW instruction. At the end of the ISR, EALLOW can be restored by the IRET instruction.

LOOP (Bit 5)— Loop instruction status bit.

LOOP is set when a loop instruction (LOOPNZ or LOOPZ) reaches the decode 2 phase of the pipeline. The loop instruction does not end until a specified condition is met. When the condition is met, LOOP is cleared. LOOP is a read-only bit; it is not affected by any instruction except a loop instruction.

When the CPU services an interrupt, the current value of LOOP is saved on the stack (when ST1 is saved on the stack), and then LOOP is cleared. Upon return from the interrupt, LOOP is not restored from the stack.

SPA (Bit 4) — Stack pointer alignment bit.

SPA indicates whether the CPU has previously aligned the stack pointer to an even address by the ASP instruction:

- 0: The stack pointer has not been aligned to an even address.
- 1: The stack pointer has been aligned to an even address.

When the ASP (align stack pointer) instruction is executed, if the stack pointer (SP) points to an odd address, SP is incremented by 1 so that it points to an even address, and SPA is set. If SP already points to an even address, SP is not changed, but SPA is cleared. When the NASP (unalign stack pointer) instruction is executed, if SPA is 1, SP is decremented by 1 and SPA is cleared. If SPA is 0, SP is not changed.

At reset, SPA is cleared.

VMAP (Bit 3) — Vector map bit.

VMAP determines whether the CPU interrupt vectors (including the reset vector) are mapped to the lowest or highest addresses in program memory:

- 0: CPU interrupt vectors are mapped to the bottom of program memory, addresses 00 0000_{16}-00 $003F_{16}$.