
# NS9750, NS9775, and NS9360 Static Memory Access Limitation Application Note

# Contents

# NS9750, NS9775, and NS9360 Static Memory Access Limitation

*This application note describes a static memory access limitation and provides a hardware workaround and a software workaround.*

## Overview

The NS9750, NS9575, and NS9360 cannot toggle the static memory output enable (`st_oe_n`) signal when performing a burst read. For example, if the CPU performs a 32-bit read through the memory controller to an 8-bit static device, `st_oe_n` is asserted for all four-byte transfers. This is an issue only with external peripherals that require an acknowledgement type strobe for each data unit transfer. Standard NOR flash and SRAM memories are not affected. NAND flash memories are affected if bursting is required.

Two workarounds available: one through software and one through external hardware. The next two sections describe the workarounds.

## Software workaround

Software can be modified to issue single accesses of the same data width as the external peripheral being accessed. For example, if the external peripheral is 8-bits wide, software must issue only single byte reads and writes to the external peripheral.

## Hardware workaround

The hardware workaround is to implement a counter based on the number of programmed wait states in an external CPLD or FPGA to generate a *byte strobe* signal.

This example shows an external device that requires four wait states:

1. Set the output enable delay (`WOEN`) to `0`.
2. Create a counter based on the read delay setting (`WTRD`).

Here is the logic and a timing diagram. In this example, the value programmed to `WTRD` is `3`, where wait = `WTRD + 4`, or four `clk_out` cycles.