

A typical TWIM write sequence including clock stretching performed by TWIM following a SUSPEND task is shown in the following figure.

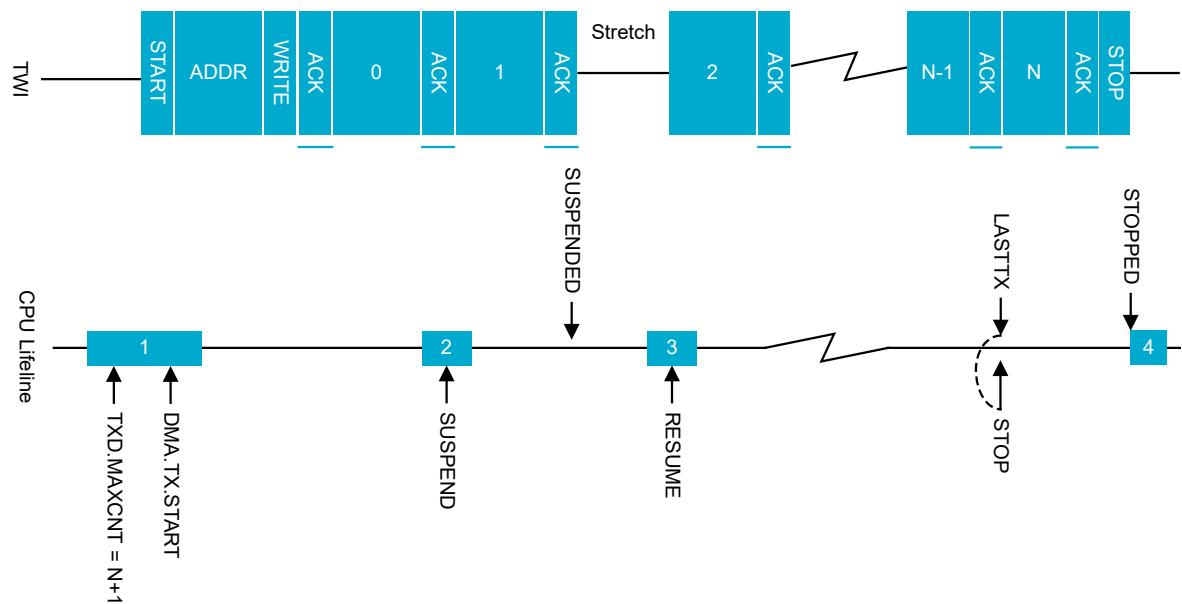


Figure 146: TWIM writing data to a target

A SUSPENDED event indicates that the SUSPEND task has taken effect.

TWIM will generate a LASTTX event when it starts to transmit the last byte.

TWIM is stopped by triggering the STOP task. To stop TWIM as fast as possible, trigger the task during the transmission of the last byte. The shortcut between LASTTX and STOP can also be used to accomplish this.

TWIM does not stop on its own when the entire RAM buffer has been sent or when an error occurs. The STOP task must be issued, either through the local or PPI shortcut, or in software as part of the error handler.

8.23.5 TWIM read sequence

A TWIM read sequence is started by triggering the DMA.RX.START task. After the DMA.RX.START task has been triggered, TWIM generates a start condition on the TWI bus. This is followed by clocking out the address and the READ/WRITE bit set to 1 (WRITE = 0, READ = 1). The address must match the address of the target device that the controller wants to read from. The READ/WRITE bit is followed by an ACK/NACK bit (ACK = 0 or NACK = 1) generated by the target.

After sending the ACK bit, the TWI target sends data to the controller using the clock generated by TWIM.

Data received will be stored in RAM at the address specified in the RXD.PTR register. TWIM will generate an ACK before the last byte is received from the target. TWIM generates a NACK after the last byte received to indicate that the read sequence will stop.

A typical TWIM read sequence is illustrated in the following figure, including clock stretching performed by TWIM following a SUSPEND task.