

6. General SMBus/I²C Architecture

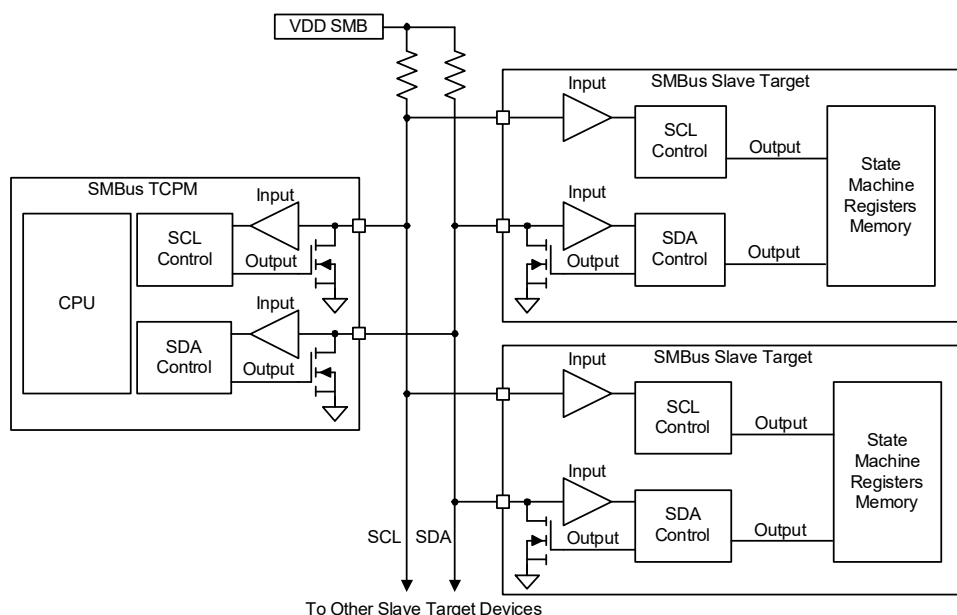


Figure 14. General SMBus Architecture

6.1 Data Validity

The data on the SDA line must be stable during the HIGH period of the SCL unless generating a START or STOP condition. The HIGH or LOW state of the data line can change only when the clock signal on the SCL line is LOW. See [Figure 15](#).

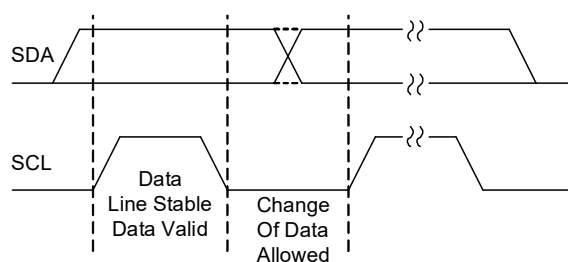


Figure 15. Data Validity

6.2 START and STOP Conditions

[Figure 16](#) shows that the START condition is a HIGH to LOW transition of the SDA line while SCL is HIGH.

The STOP condition is a LOW to HIGH transition on the SDA line while SCL is HIGH. A STOP condition must be sent before each START condition.

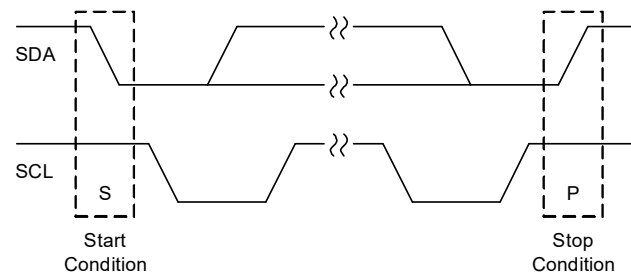


Figure 16. Start and Stop Waveforms

6.3 Acknowledge (ACK)

Each address and data transmission uses nine clock pulses. The ninth pulse is the Acknowledge bit (ACK). After the start condition, the TCPM sends seven slave target address bits and a R/W bit during the next eight clock pulses. During the ninth clock pulse, the device that recognizes its own address holds the data line LOW to acknowledge (see Figure 17). Both the TCPM and slave target use the ACK bit to acknowledge receipt of register addresses and data.

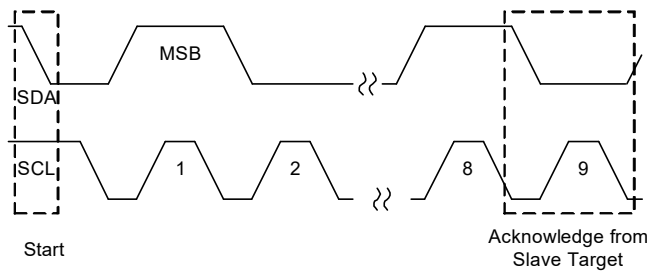


Figure 17. Acknowledge on the SMBus

6.4 Writing Single Byte Registers

When writing to a single byte register, use the following transaction.

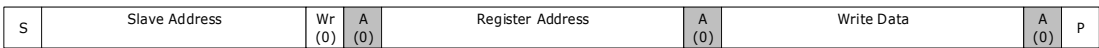


Figure 18. Writing Single Byte Registers

6.5 Reading Single Byte Registers

When reading to a single byte register, use the following transaction.

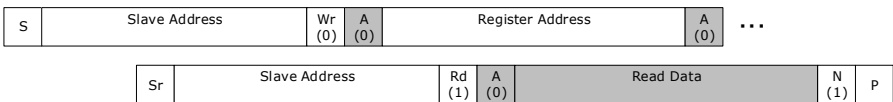


Figure 19. Reading Single Byte Registers

6.6 Writing Two-Byte Registers

When writing to a two-byte register, use the following transaction.

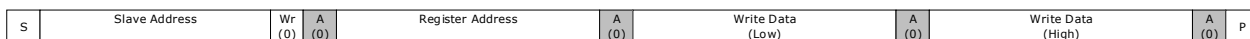


Figure 20. Writing Two-Byte Registers

6.7 Reading Two-Byte Registers

When reading a two-byte register, use the following transaction.

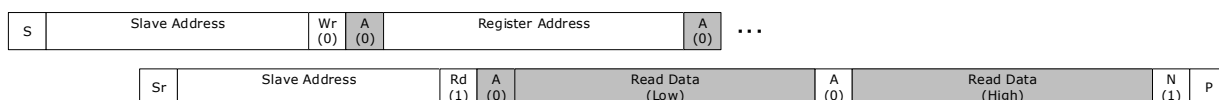


Figure 21. Reading Two-Byte Registers

6.8 Writing the TRANSMIT_BUFFER

When writing to the TRANSMIT_BUFFER register, use the following transaction.

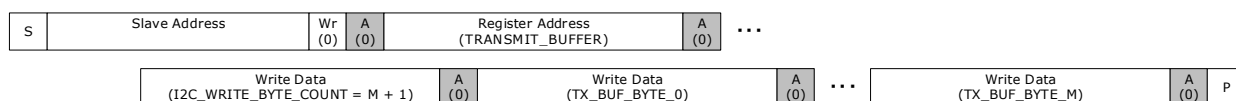


Figure 22. Writing the TRANSMIT_BUFFER

6.9 Reading the RECEIVE_BUFFER

When reading the RECEIVE_BUFFER register, use the following transaction.

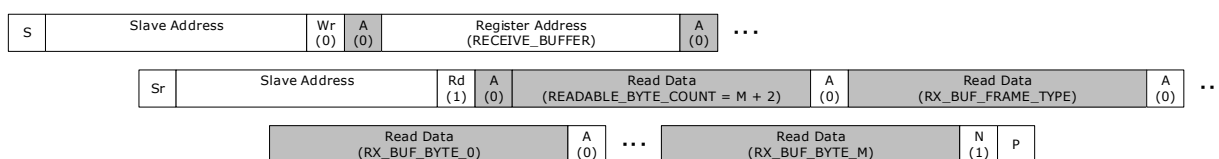


Figure 23. Reading the RECEIVE_BUFFER

6.10 Reading the Alert Response Address

When reading the Alert Response Address register, use the following transaction.

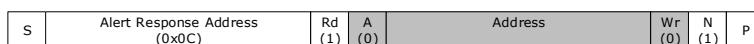


Figure 24. Reading the Alert Response Address