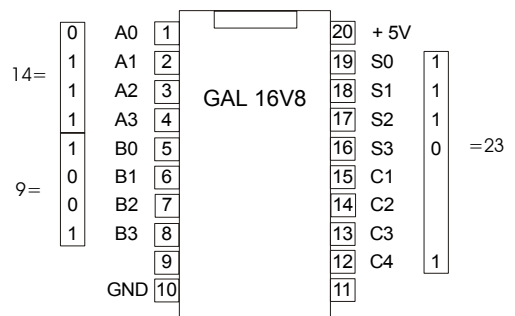


## 4-Bit-Volladdierer

Das GAL wird als **4-Bit Volladdierer** eingesetzt.

Einen Volladdierer verwendet man, wenn man mehrstellige Dualzahlen addieren möchte. Dieser muss in der Lage sein zwei zueinander gehörende Ziffern der beiden Dualzahlen und den Übertrag der nächst niederen Stelle zu verarbeiten. Die beiden Binärzahlen, die addiert werden sollen, sind die Bits A0...A3 und B0...B3. Bei den **Logikgleichungen** im GAL16V8 muss beachtet werden, dass für die Übergänge C1...C3 rückgekoppelte Ausgänge vorzusehen sind.



Die **allgemeine Logikgleichung** für den Volladdierer ist:

$$\begin{aligned} S_n &= \neg A_n \cdot B_n \cdot \neg C_n + A_n \cdot \neg B_n \cdot \neg C_n + \neg A_n \cdot B_n \cdot C_n + A_n \cdot B_n \cdot C_n \\ C_{n+1} &= A_n \cdot B_n + A_n \cdot C_n + B_n \cdot C_n \end{aligned}$$

CHIP ADD4BIT GAL16V8 COMPLEX\_MODE

| A0 | A1 | A2 | A3 | B0 | B1 | B2 | B3 | NC | GND |
|----|----|----|----|----|----|----|----|----|-----|
| NC | C4 | C3 | C2 | C1 | S3 | S2 | S1 | S0 | VCC |

$$\begin{aligned} S0 &= \neg A0 \cdot B0 \\ &+ A0 \cdot \neg B0; \end{aligned}$$

$$C1 = A0 \cdot B0$$

$$\begin{aligned} S1 &= A1 \cdot B1 \cdot \neg C1 \\ &+ \neg A1 \cdot \neg B1 \cdot C1 \\ &+ \neg A1 \cdot B1 \cdot C1 \\ &+ A1 \cdot \neg B1 \cdot C1; \end{aligned}$$

$$\begin{aligned} C2 &= A1 \cdot B1 \\ &+ A1 \cdot C1 \\ &+ B1 \cdot C1; \end{aligned}$$

$$\begin{aligned} S2 &= A2 \cdot B2 \cdot \neg C2 \\ &+ \neg A2 \cdot \neg B2 \cdot C2 \\ &+ \neg A2 \cdot B2 \cdot C2 \\ &+ A2 \cdot \neg B2 \cdot C2; \end{aligned}$$

$$\begin{aligned} C3 &= A2 \cdot B2 \\ &+ A2 \cdot C2 \\ &+ B2 \cdot C2; \end{aligned}$$

$$\begin{aligned} S3 &= A3 \cdot B3 \cdot \neg C3 \\ &+ \neg A3 \cdot \neg B3 \cdot C3 \\ &+ \neg A3 \cdot B3 \cdot C3 \\ &+ A3 \cdot \neg B3 \cdot C3; \end{aligned}$$

$$\begin{aligned} C4 &= A3 \cdot B3 \\ &+ A3 \cdot C3 \\ &+ B3 \cdot C3; \end{aligned}$$