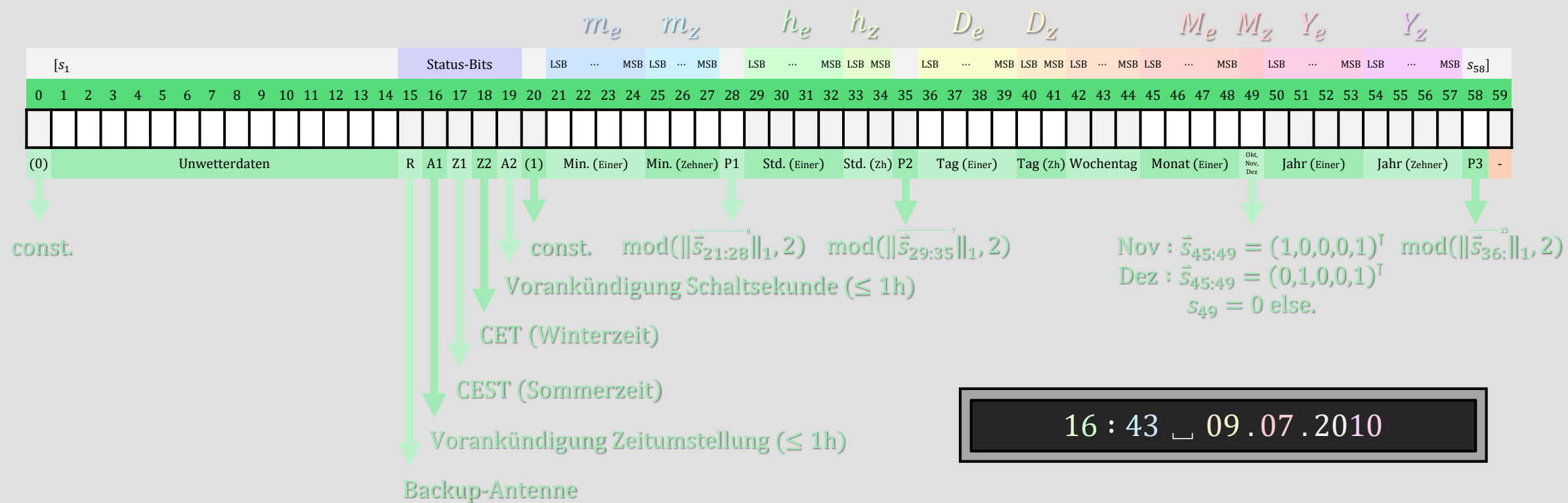


Weitere Übungen zu Bit- und Schiebeoperatoren

- ① Am **09.07.** im Jahre **2010** lag um **16:43** Uhr bei dem deutschen Zeitzeichensender **DCF77** eine Störung vor. Zwar konnten Datum und Uhrzeit korrekt übermittelt werden, jedoch war das Feld für die Unwetterdaten in der ausgestrahlten Sequenz vollständig genullt. Weiter konnte der Sequenz die Inbetriebnahme der Backup-Antenne entnommen werden.

Im Folgenden sei die vollständige Zeitzeichensequenz aus den vorliegenden Informationen zu gewinnen und in das vorliegende Schema in korrekter Bitordnung einzutragen:



- ② Es sei Ausgabe und Arbeitsweise des folgenden Programmsnipsels zu untersuchen:

```
uint8_t x = 0b00100110; // 0x26
uint8_t y = 0b01010011; // 0x53
x = x ^ y;
y = x ^ y;
x = x ^ y;
printf("0x%02X_0x%02X\n\r", x, y);
```

$$\theta x \boxed{} \boxed{} \cup \theta x \boxed{} \boxed{} \leftarrow \text{ } \quad \leftarrow \text{ }$$

- ③ Der folgende Code zeigt die Implementierung der globalen Übergangsfunktion eines zellulären Automaten $\mathcal{A} := \langle \Omega, Q, \dot{\mathbf{U}}_{\text{TD}}, \delta \rangle$ in einem zweidimensionalen Rechteckraster:

```
bool rule[16] = {0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0};

void update() {
    bool new_grid[GRID_SIZE][GRID_SIZE] = {0};    // GRID_SIZE = 31
    for (int y = 0; y < GRID_SIZE; ++y) {
        for (int x = 0; x < GRID_SIZE; ++x) {
            bool S = grid[x][y];
            bool TL = (y+1 < GRID_SIZE && x < GRID_SIZE-1) ? grid[x+1][y+1] : false;
            bool T = (y+1 < GRID_SIZE) ? grid[x][y+1] : false;
            bool TR = (y+1 < GRID_SIZE && x > 0) ? grid[x-1][y+1] : false;
            int idx = (TL << 3) | (T << 2) | (TR << 1) | S;
            bool next = rule[idx];
            next = next || S;
            new_grid[x][y] = next;
        }
    }

    memcpy(grid, new_grid, sizeof(grid));
}
```

$\mathcal{A}$ sei für 7 weitere Generationen (Ausführung von **update**) zu simulieren. Lebendige Zellen sind zu kennzeichnen. Der Garten Eden enthält genau eine lebendige Zelle:

