

# Konzeption, Bau und Programmierung einer lateinischen Wortuhr

Verfasst von Tamara Schneider

im Fach Informatik

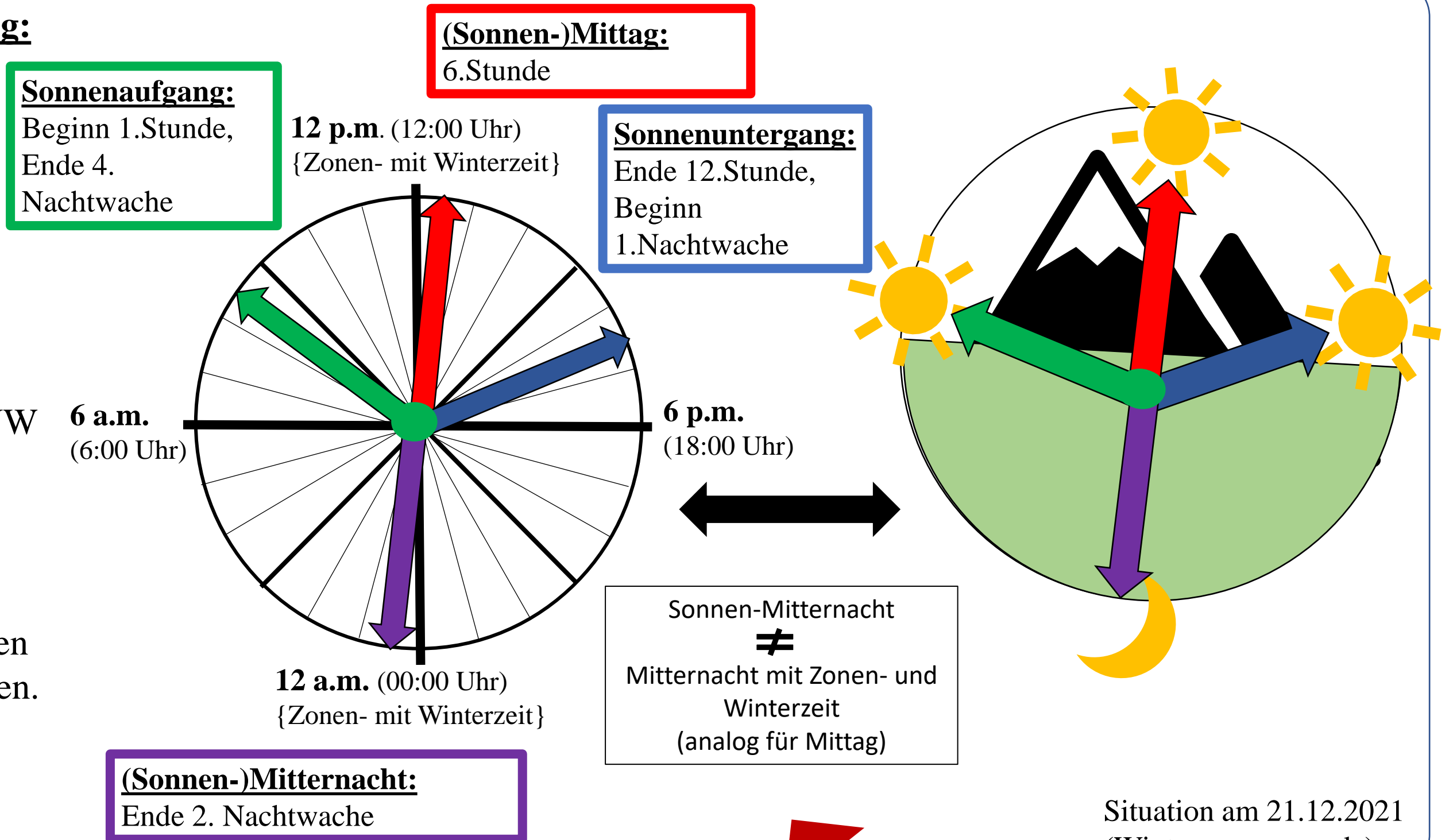
betreut von Chris Weber

## Modus 2 – lateinische Stundenzählung:

**Tag (12 h)** → Zählung in 12 Stunden:  
1. Stunde beginnt bei Sonnenaufgang,  
6. Stunde um Mittag  
(Sonnenhöchststand), 12. Stunde endet  
bei Sonnenuntergang

**Nacht (12h)** → Zählung in 4  
Nachtwachen (NW) à je 3 Stunden: 1. NW  
beginnt bei Sonnenuntergang, 2. NW  
endet um (Sonnen-)Mitternacht, 4. NW  
endet bei Sonnenaufgang

→ Die «römischen Stunden» entsprechen  
**nicht(!)** unseren Stunden von 60 Minuten.  
Die «Stunden» variieren in ihrer Dauer  
durch die unterschiedlichen Tages- und  
Nachtlängen.



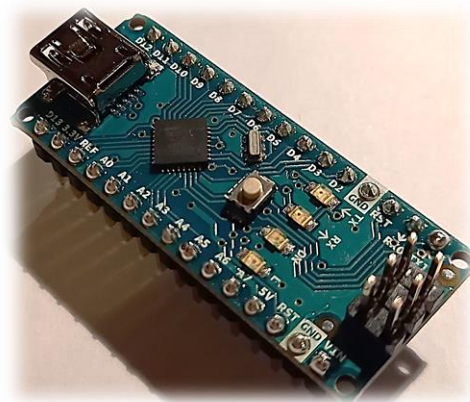
## Modus 1 – lateinisch formulierte Zeit:

Entspricht exakt unserer Uhrzeit (Zonenzeit mit  
Winter- oder Sommerzeit)

**Bemerkung:** 12-Stundeneinteilung mit Vormittag  
(a.m.) und Nachmittag (p.m.) als Anhängsel

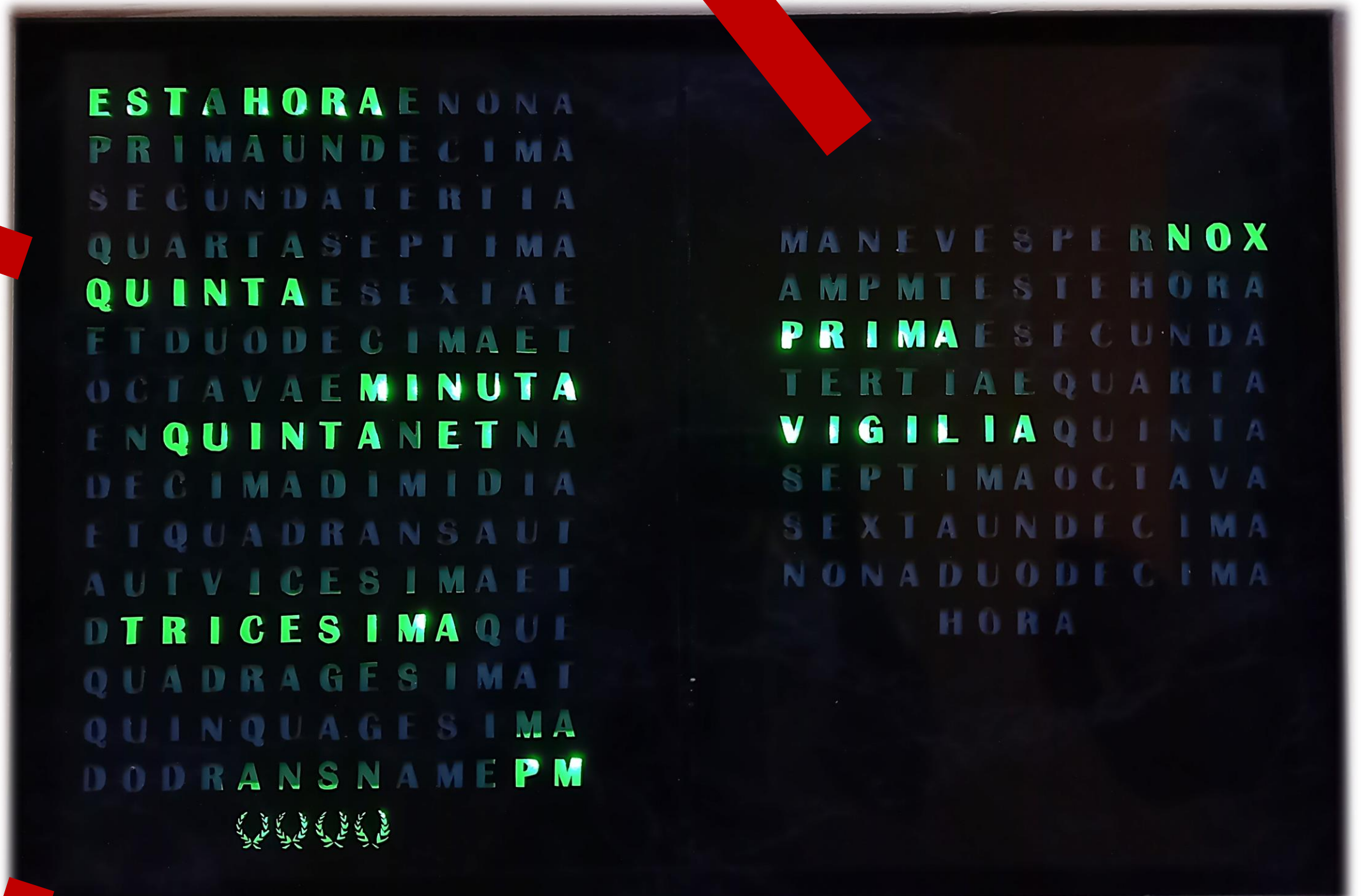
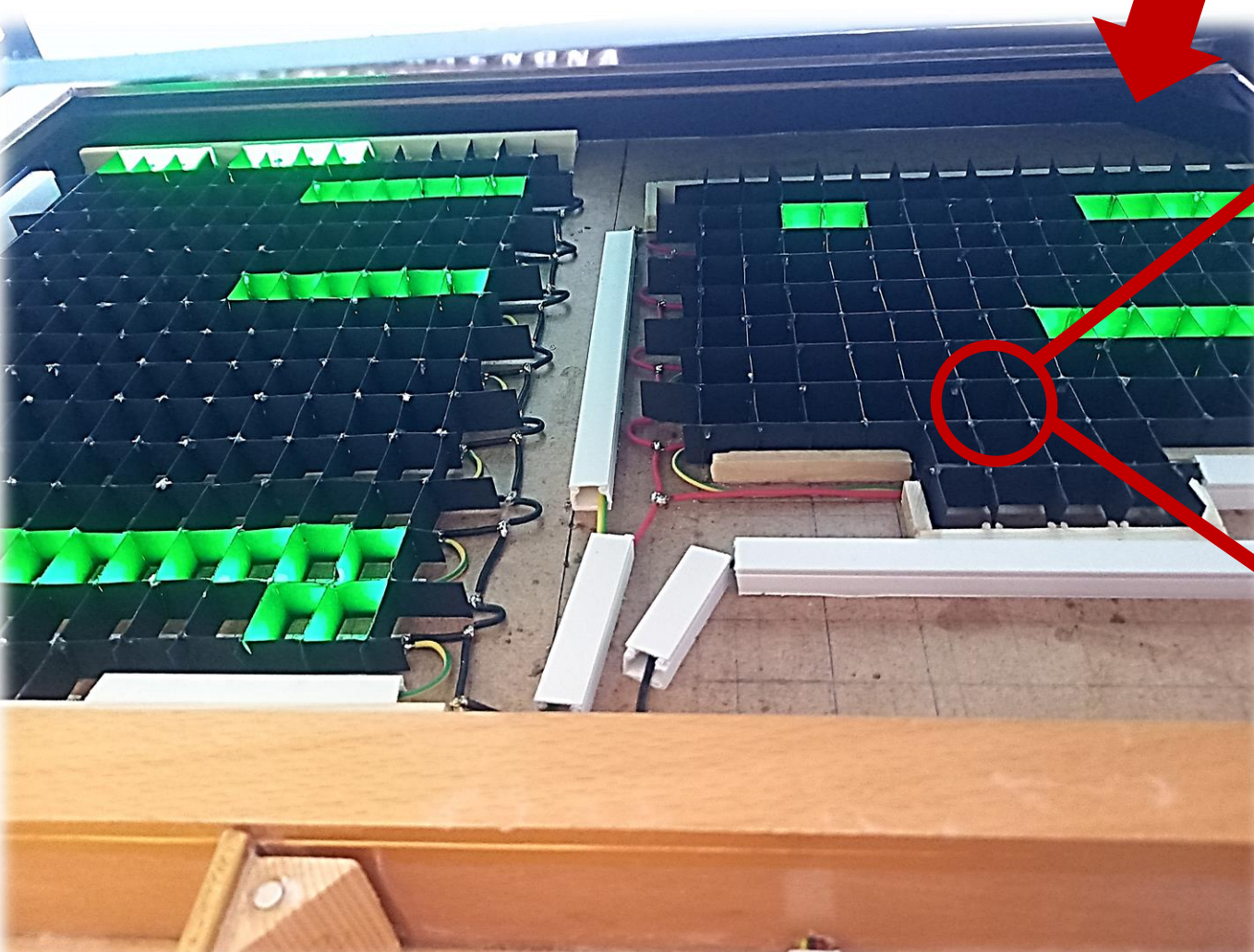
## Programmierung:

Programmiersprache: C++  
Arduino IDE  
→ (Integrated Development  
Environment)  
Mikrocontroller: Arduino Nano



Alle Komponenten wurden zuerst einzeln  
getestet.

Zur Erleichterung des Testens wurde ein Prototyp  
angefertigt, der dieselben Funktionen aufweist  
wie das Endprodukt



**Übersetzung:**  
*Es ist fünf Uhr 39 nachmittags* → (17:39 Uhr)

**Übersetzung:**  
*Nacht. Es ist die erste Nachtwache*

### Ein Blick ins Innere:

Matrix bestehend aus **Neopixel**  
→ Vorteile: Am Streifen  
erhältlich und können alle jeweils  
einzeln angesteuert werden. →  
Erhebliche Vereinfachung der  
Programmierung

**Raster** zur Unterteilung und  
Abgrenzung der einzelnen  
Neopixel voneinander

