

Make Your ESP32 Knowledgeable About Date/Time

```
LINE_messaging_different_reactions.ino
72 }
73 else {
74     sprintf(outgoingPacket, "[%s report] %s boiler: critical, %.1f°C; call the boss", nodeID, currentTime, x);
75     LINE.setToken(token[2]);
76     LINE.notifyPicture(outgoingPacket, "https://shorturl.at/etAS3");
77 }
78 Serial.printf("Message sent: %s\n", outgoingPacket);
79
80 if (timeinfo.tm_hour <= 7 || timeinfo.tm_hour >= 18)
81     delay(30000); //During non-working hours, report the readings every half a minute
82 else
83     delay(20000);
84 }
85
86 void getLocalTime() {
87     //struct tm timeinfo;
88
89     if (!getLocalTime(&timeinfo)){
90         Serial.println("Failed to obtain time");
91         return;
92     }
93
94     //strftime(currentTime, 80, "%b/%d/%Y %H:%M:%S", &timeinfo);
95     strftime(currentTime, 80, "%Y-%m-%d %H:%M:%S", &timeinfo); //see https://cplusplus.com/reference/ctime/strftime/
96     //Serial.printf("Current time: %s\n", currentTime);
97     //Serial.println(&timeinfo, "%b/%d/%Y %H:%M:%S");
98     //Serial.printf("Today: %s\tcurrent hour: %d\n", whichDay[timeinfo.tm_wday], timeinfo.tm_hour);
99 }
```

Output

Ln 95, Col 58 WEMOS LOLIN32 on COM4 [not connected]



While a typical micro-controller does not have exact knowledge of date and time, an ESP32 is able to work smarter for energy saving application during daytime or nighttime, during weekdays or weekends. Your ESP32 can be programmed to switch on/off appliances if it is not the scheduled time for them to operate.