

Lab 3: Radio-controlled Clock with DCF77

Dokumentation

Tim Jauch

Ergün Bickici

Contents

1	Requirements	5
2	User Interface Description	6
2.1	LCD Display	6
2.2	LED Indicators	6
2.3	Control Button	6
3	Debugging	7
4	Data Dictionary	8
4.1	List of Global Variables	8
4.2	Hardware Resources	8
5	Module overview	9
6	Interface Descriptions of All Subroutines	10
6.1	main.c	10
6.2	clock.c	10
6.3	dcf77.c	10
6.4	lcd.asm	11
6.5	led.asm	11
6.6	button.asm	11
7	Flow Charts	12
7.1	Main	12
7.2	Clock	13
7.3	DCF77	15

List of Figures

1	Module Overview	9
2	Main	12
3	Clock 1	13
4	Clock 2	14
5	DCF77 - Part 1	15
6	DCF77 - Part 2	16
7	DCF77 - Part 3	17
8	Event Minute – Part 1	18
9	Event Minute – Part 2	19
10	Event Minute – Part 3	20
11	Event Minute – Part 4	21
12	Event Minute – Part 5	22
13	USDE Pressed	23
14	isLeap	24
15	daysInMonth	25
16	adjustDate	26
17	USDE	27

List of Tables

1	Global variables used across modules	8
2	Hardware resources accessed by software modules	8

1 Requirements

- The first line of the Dragon12's LCD shall display the time in the format: `hh:mm:ss`. The second line shall show the date in the format: `dd.mm.yyyy`. Manual time setting, as implemented in Lab 2, is no longer required.
- Since DCF77 signal reception may occasionally fail (e.g., due to poor radio conditions in reinforced concrete buildings or during transmitter maintenance), the internal timer-based clock from Lab 2 remains active. The DCF77 signal is only used to adjust the time once per minute and to display the date. If the DCF77 data becomes unavailable, the system continues running solely on the internal timer.
- The internal timer clock shall toggle the LED connected to Port B.0 once per second.
- The LED on Port B.1 shall reflect the raw DCF77 signal: it turns on when the signal is low, and off when it is high. As a result, this LED will toggle once per second when the DCF77 signal is being received.
- The LED on Port B.3 shall be turned on when a complete and valid DCF77 time and date message has been successfully decoded. It turns off immediately if no valid data is received or the data is corrupted.
- The DCF77 signal is connected to Port H.0 on the real Dragon12 board. Due to potential signal spikes and jittering edges, it is recommended to poll the signal every 10 ms instead of relying on Port H interrupts.

2 User Interface Description

2.1 LCD Display

- The first line of the Dragon12 LCD displays the current time in the format `hh:mm:ss`.
- The second line shows the date in the format `dd.mm.yyyy`.
- The day of the week (e.g., `Sun`, `Mon`, ...) is displayed in front of the date.
- The active time zone is indicated by either `DE` or `US` on the LCD.

2.2 LED Indicators

- The LED connected to Port B.0 toggles once per second, driven by the internal timer clock.
- The LED on Port B.1 reflects the raw DCF77 signal level: it turns on when the signal is low and off when the signal is high. This results in a 1 Hz blinking pattern during proper DCF77 reception.
- The LED on Port B.2 lights up when an error is detected and remains on until valid data is received.
- The LED on Port B.3 turns on when a complete and valid DCF77 time and date frame is successfully decoded. It turns off immediately if the received data is invalid or missing.

2.3 Control Button

- The button connected to PTH3 toggles between the MET and EST time zones.

3 Debugging

No debugging was required, as all provided code segments functioned correctly. Only the necessary extensions to the `dcf77.c` and `clock.c` files had to be implemented.

4 Data Dictionary

4.1 List of Global Variables

Module(s)	Variable	Type	Purpose
clock.c, main.c	clockEvent	enum	Defines clock event types (e.g., SET, TICK, ALARM)
clock.c, dcf77.c	dcf77TimeZone	enum	Indicates selected time zone: MET or EST
clock.c, dcf77.c, main.c	dcf77Event	enum	Represents different signal states or transitions from the DCF77 decoder

Table 1: Global variables used across modules

4.2 Hardware Resources

Module	HCS12 Hardware	Purpose
dcf77.c	Port H	Receives the DCF77 antenna signal via input pin
led.asm	Port B	Controls status LEDs (DCF77 signal, timer pulses, error states)
lcd.asm	Dragon12 LCD	Displays time, date, weekday, and time zone

Table 2: Hardware resources accessed by software modules

5 Module overview

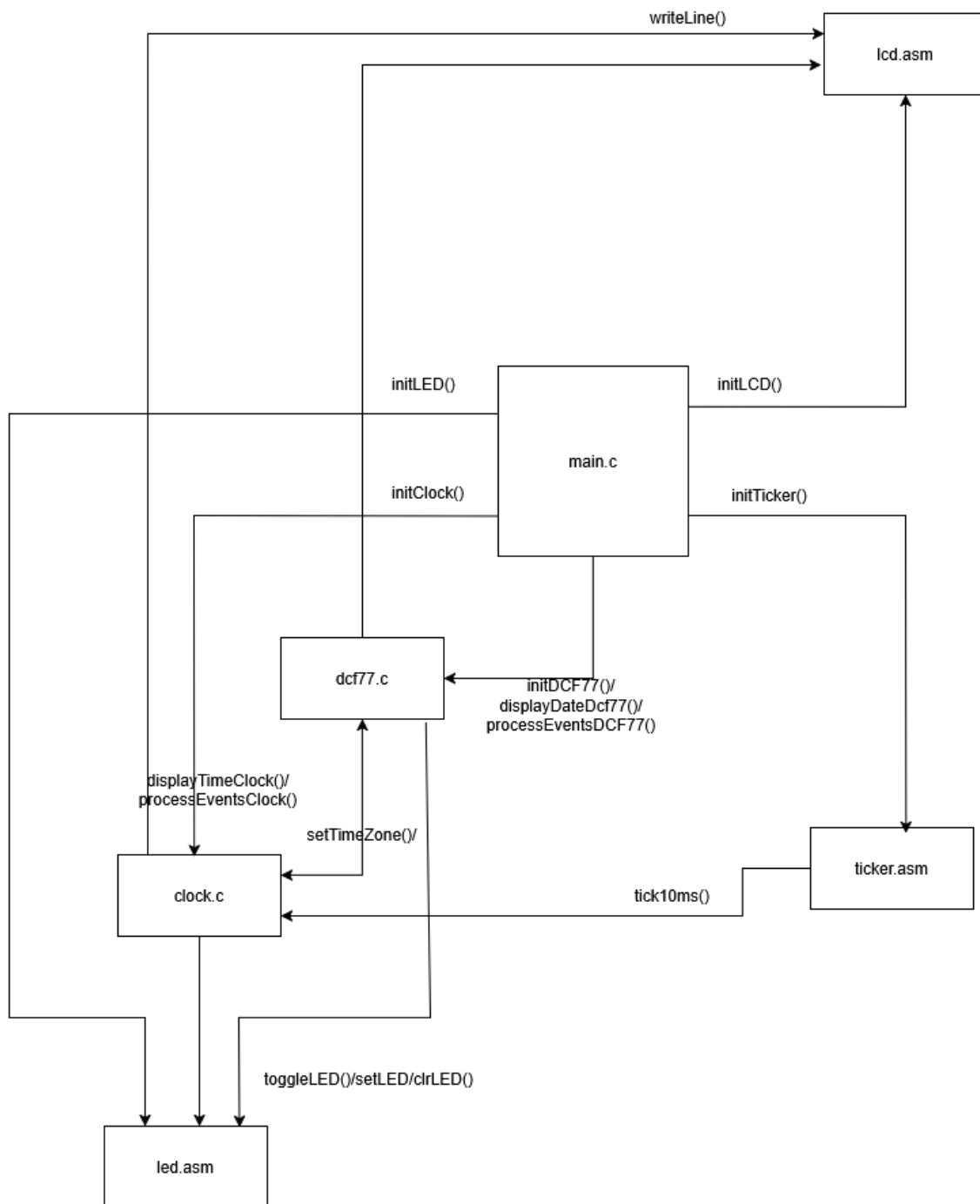


Figure 1: Module Overview

6 Interface Descriptions of All Subroutines

6.1 main.c

- **main()** Initializes all modules and calls the corresponding subroutines.

6.2 clock.c

- **initClock()** Loads the initial time and weekday.
- **tick10ms()** Called every 10 ms to advance internal clock logic.
- **processEventsClock(clockEvent)** Updates internal time based on received event.
- **setClock(hours, minutes, seconds)** Sets the current time, e.g. based on DCF77 data.
- **tzName()** Returns the time zone abbreviation.
- **tzHours()** Returns the time zone offset.
- **displayTimeClock()** Writes current time to the LCD using **writeLine()**.
- **time()** Returns the system uptime in milliseconds.

6.3 dcf77.c

- **initDCF77()** Initializes the DCF77 module and decoder state.
- **initializePort()** Configures Port H.0 for input signal polling.
- **readPort()** Reads the DCF77 signal from Port H.0 (0 = LOW, 1 = HIGH).
- **sampleSignalDCF77(currentTime)** Called every 10 ms. Measures pulse durations and returns DCF77 events.
- **processEventsDCF77(event)** Processes signal states via a DCF77 event state machine.
- **updateDate()** Updates system date and time, and refreshes the display.
- **displayDateDcf77()** Displays the decoded date on the LCD using **writeLine()**.
- **setDateDcf77()** Validates the received data and updates internal variables.
- **verifyValue(start, end)** Verifies a BCD-coded value including parity check.
- **setTimeZone(timeZone, hours, minutes)** Sets the system time zone and applies the offset.

6.4 `lcd.asm`

- **writeLine(line, text)** Displays a text string on the specified LCD line.

6.5 `led.asm`

- **initLED()** Initializes the LED output pins on Port B.
- **setLED0(on)** Toggles LED on Port B.0 every second.
- **setLED1(signal)** Reflects raw DCF77 signal on Port B.1.
- **setLED2(error)** Lights up Port B.2 if invalid data is detected.
- **setLED3(valid)** Turns on Port B.3 when a valid DCF77 frame is decoded.

6.6 `button.asm`

- **USDE_Pressed()** Detects button press on PTH3 to toggle between DE and US time zones.

7 Flow Charts

7.1 Main

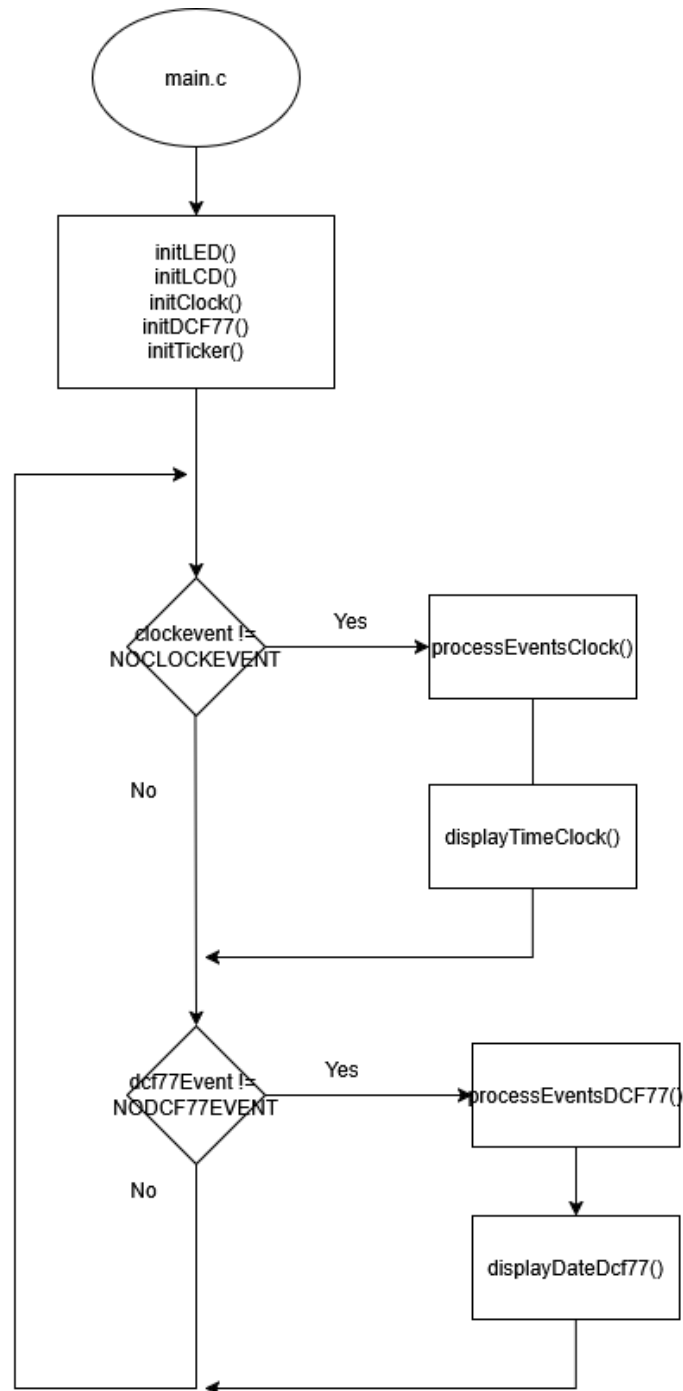


Figure 2: Main

7.2 Clock

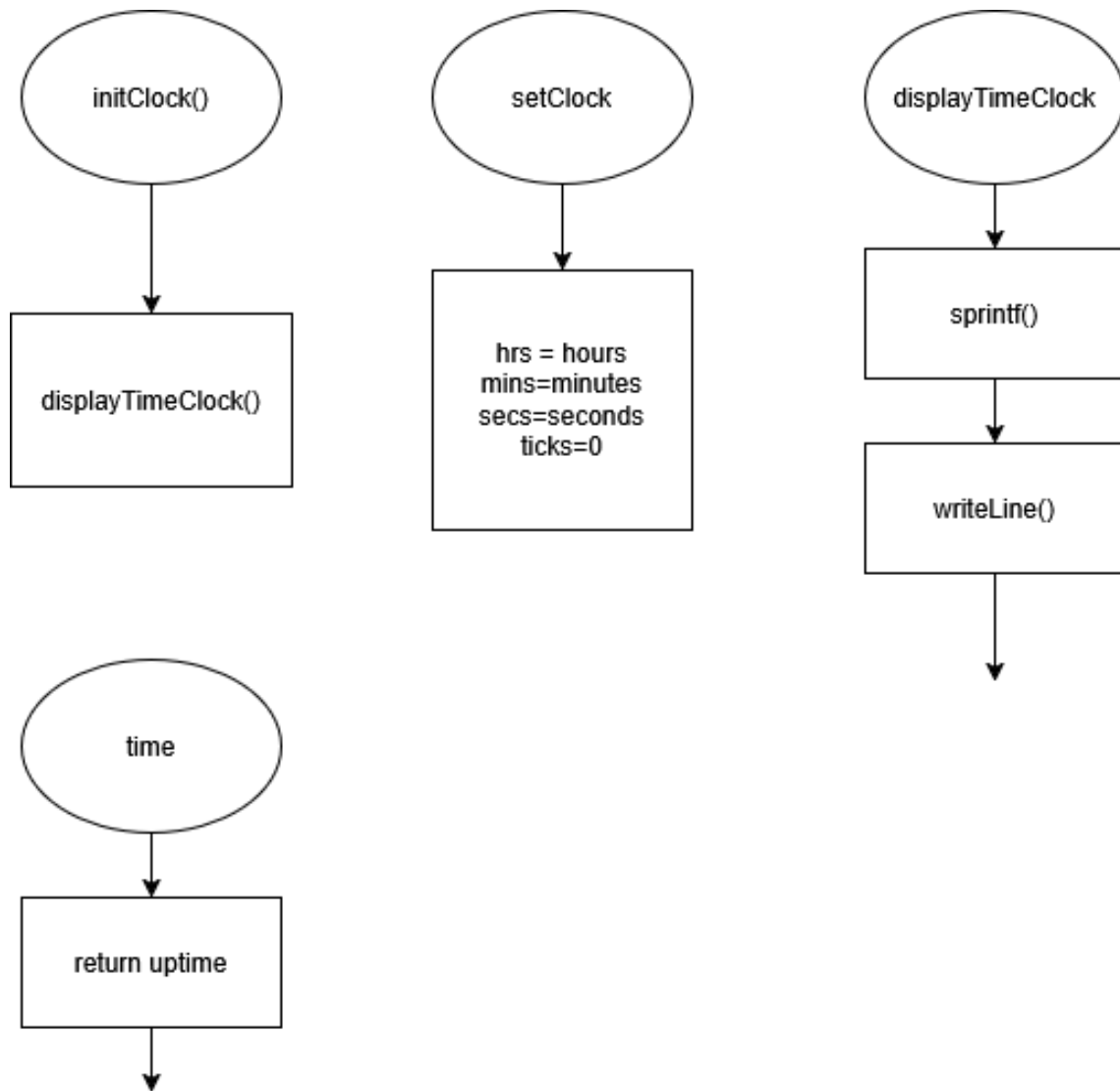


Figure 3: Clock 1

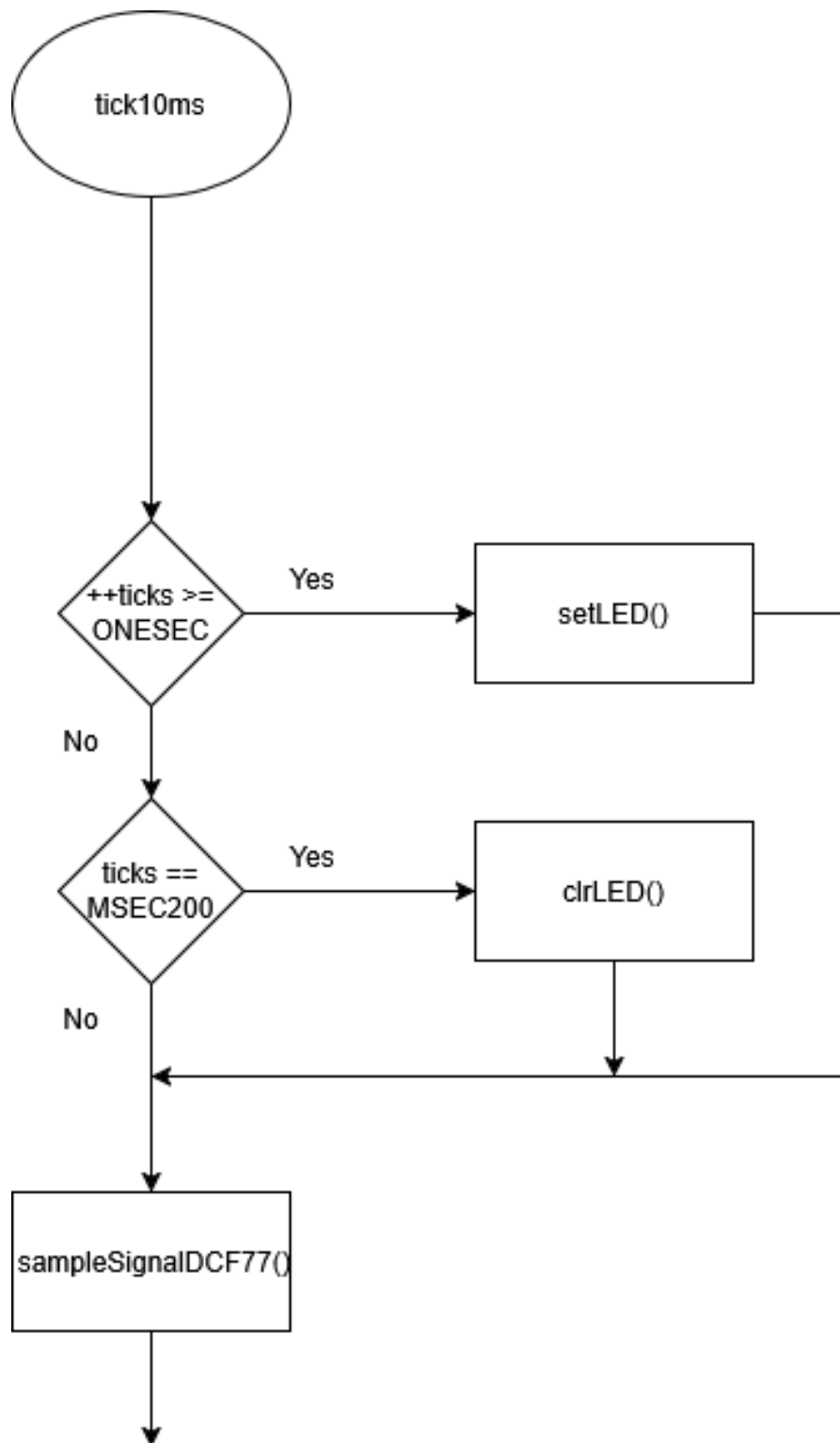


Figure 4: Clock 2

7.3 DCF77

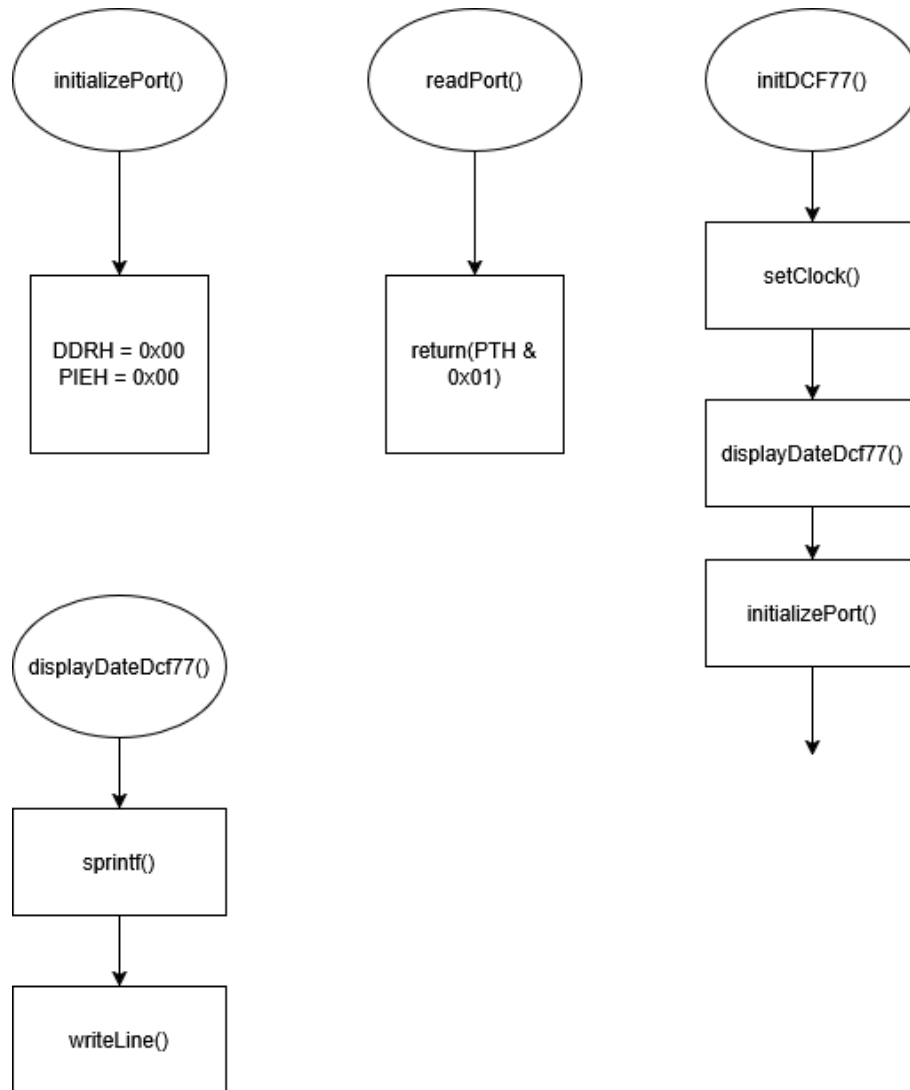


Figure 5: DCF77 - Part 1

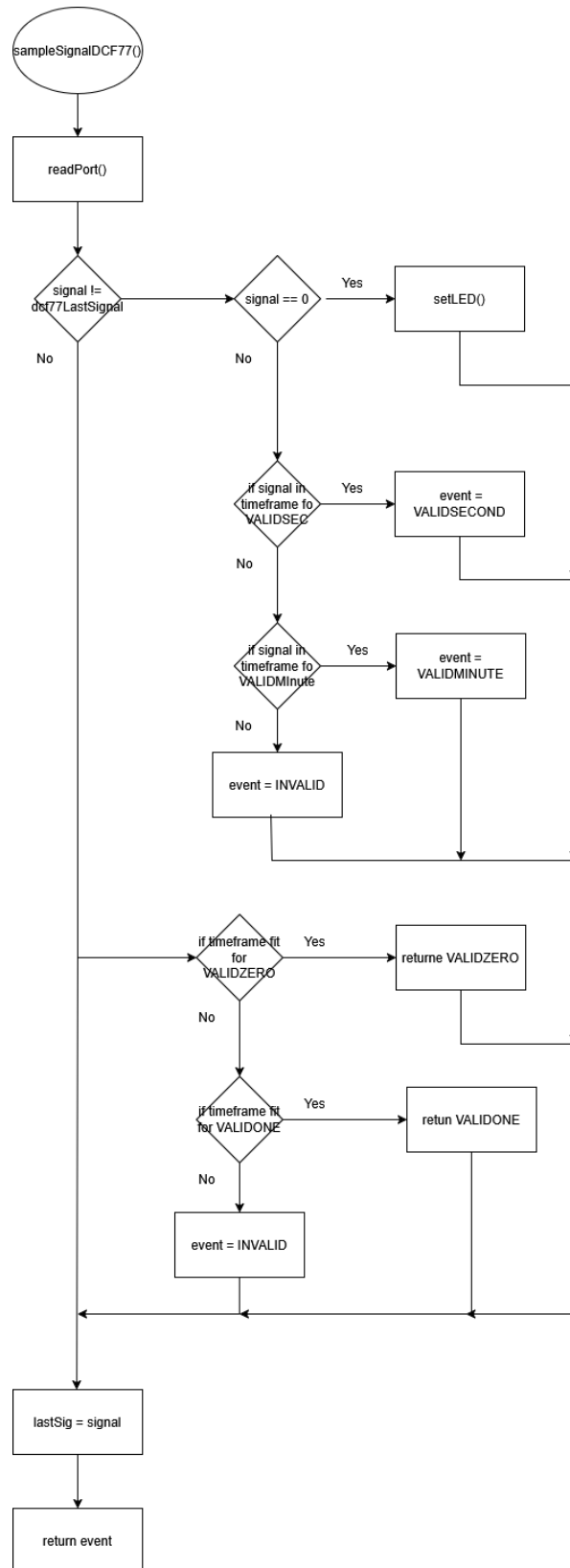


Figure 6: DCF77 - Part 2

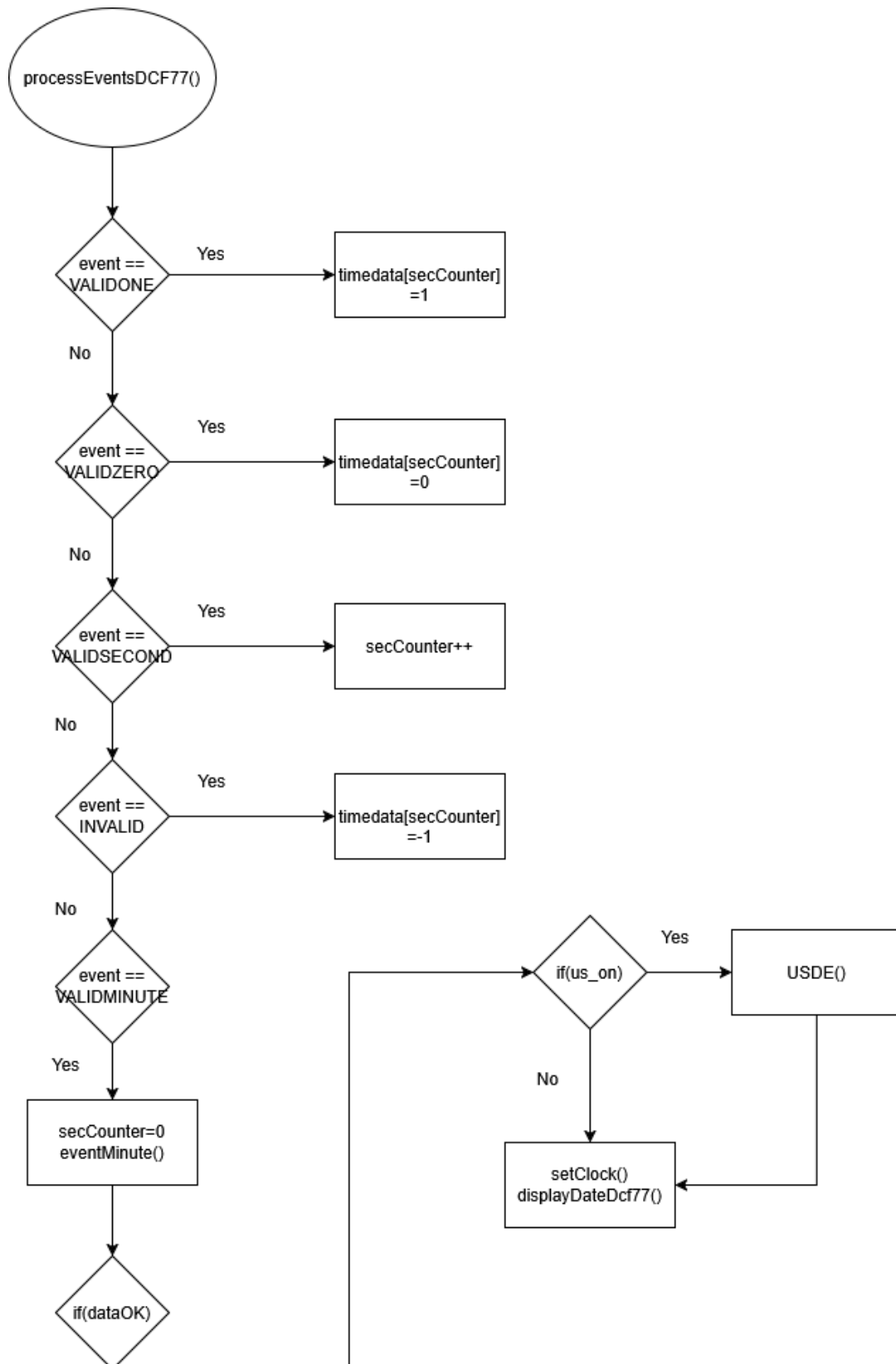


Figure 7: DCF77 - Part 3

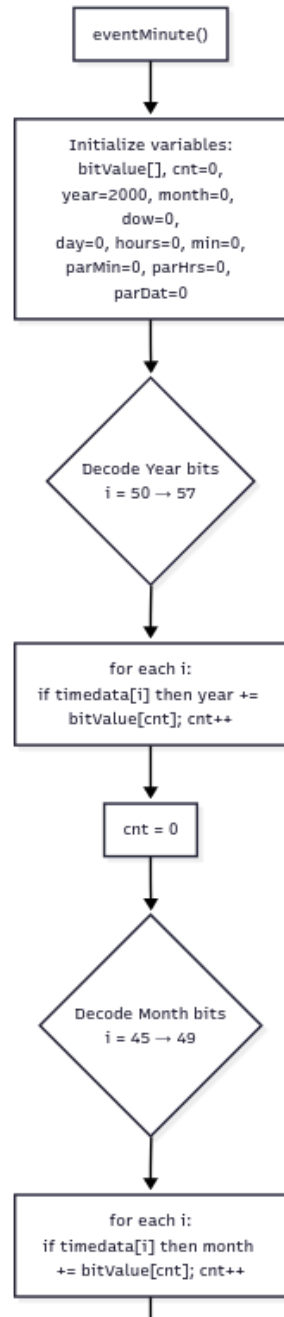


Figure 8: Event Minute – Part 1

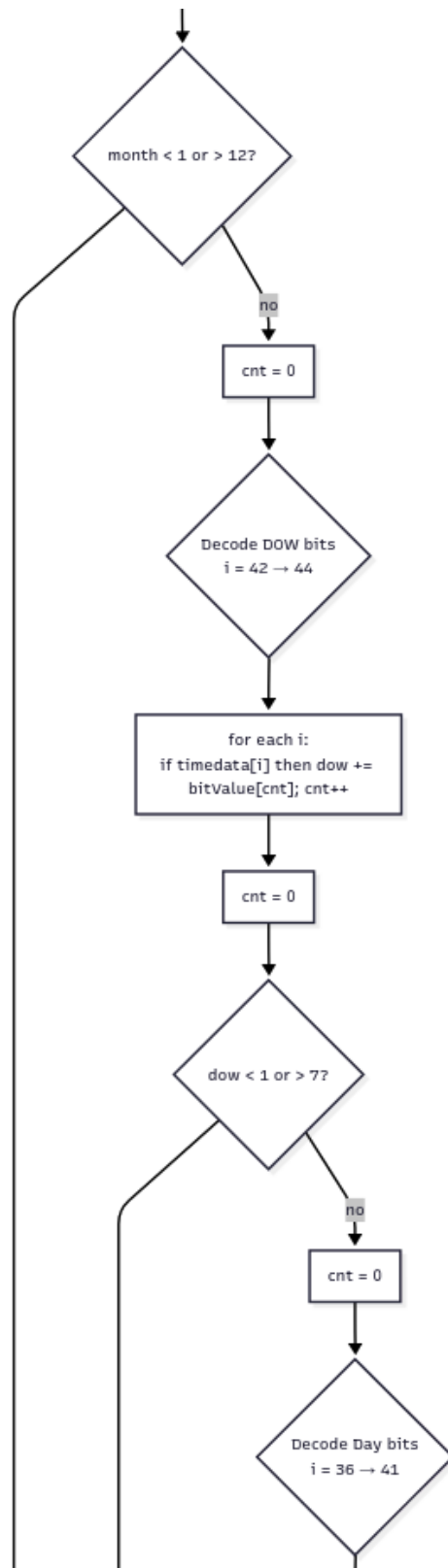


Figure 9: Event Minute – Part 2

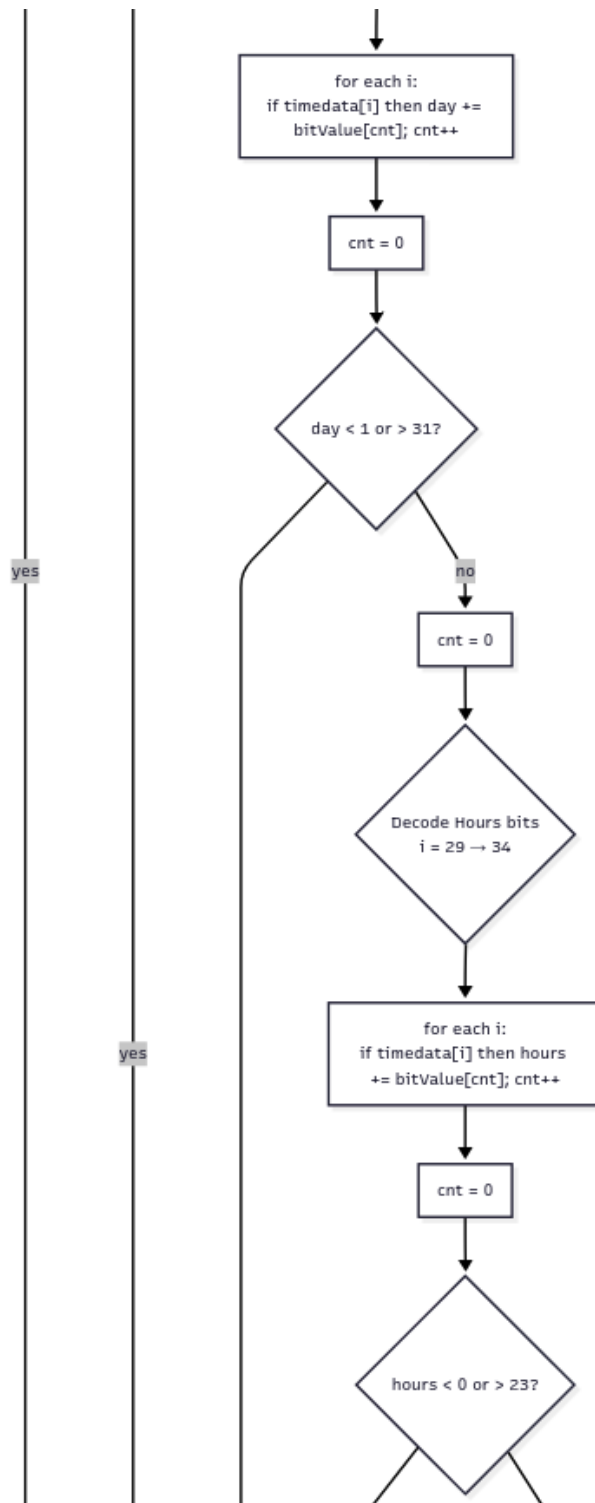


Figure 10: Event Minute – Part 3

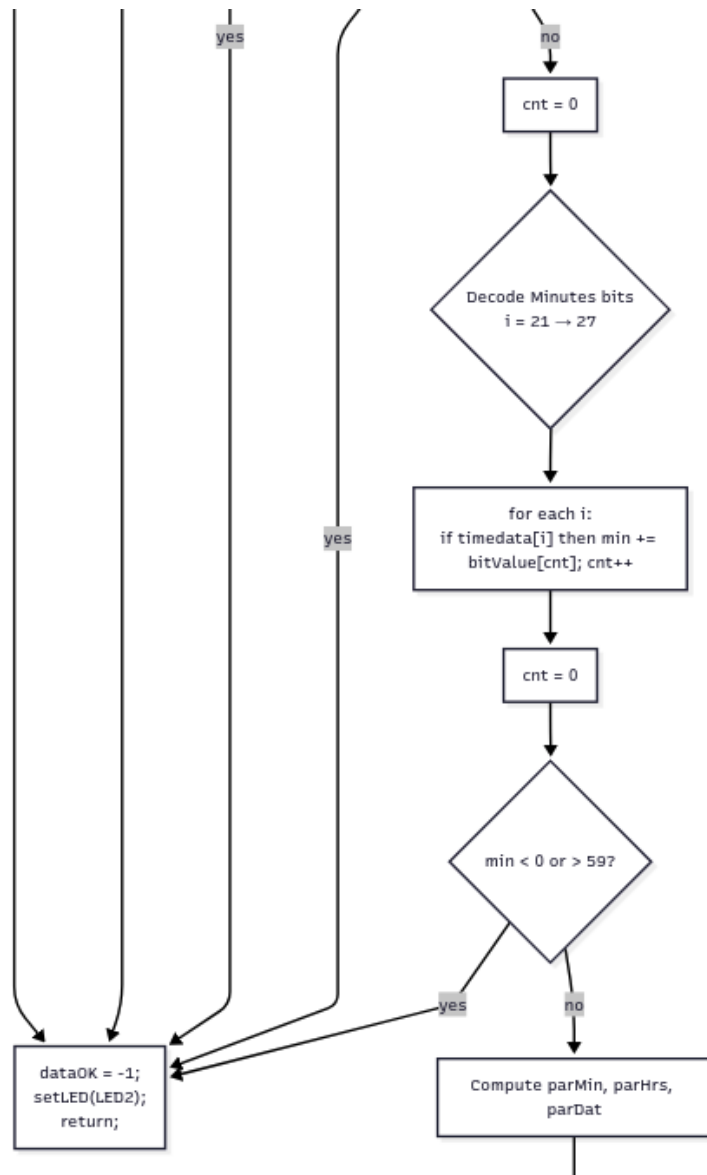


Figure 11: Event Minute – Part 4

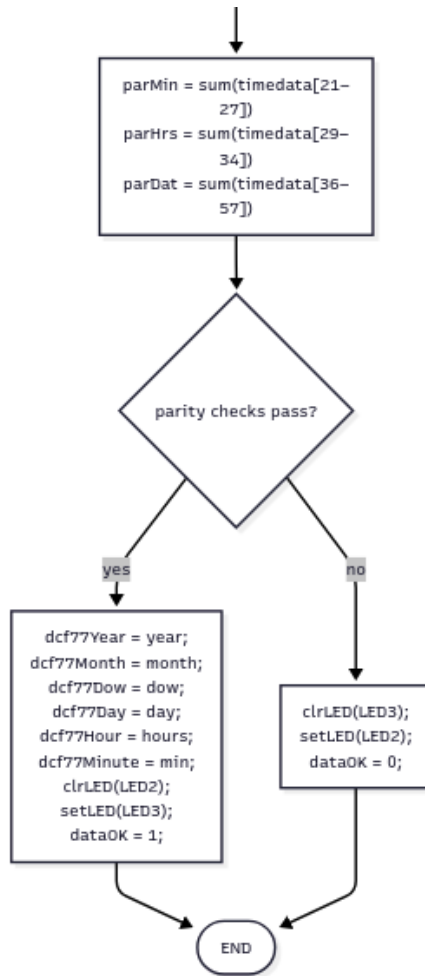


Figure 12: Event Minute – Part 5

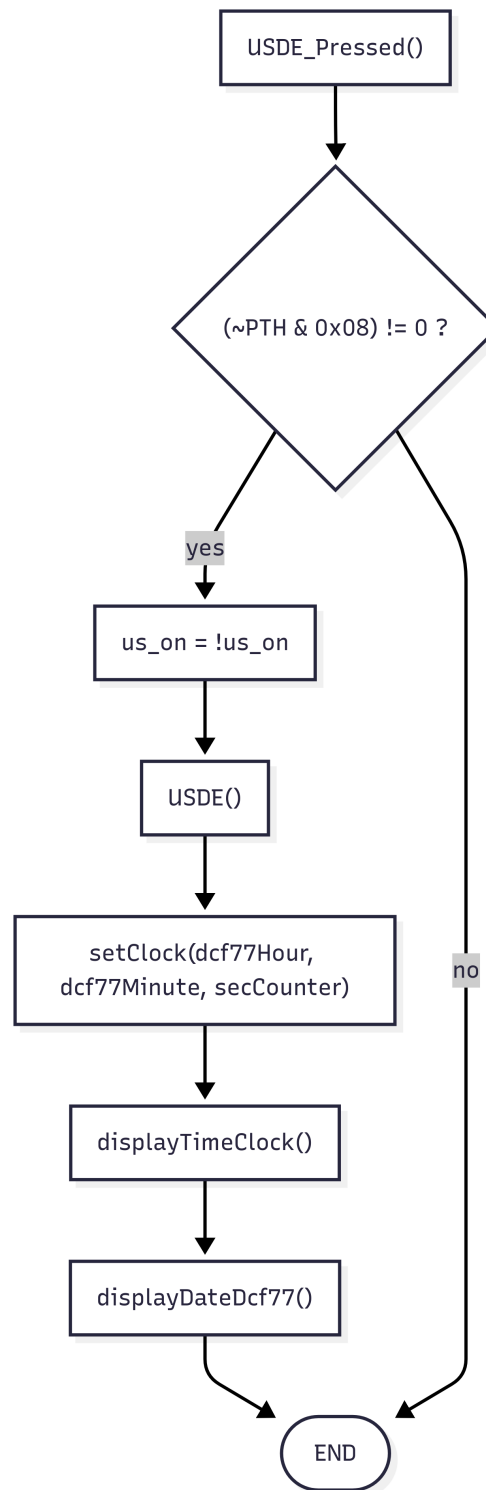


Figure 13: USDE Pressed

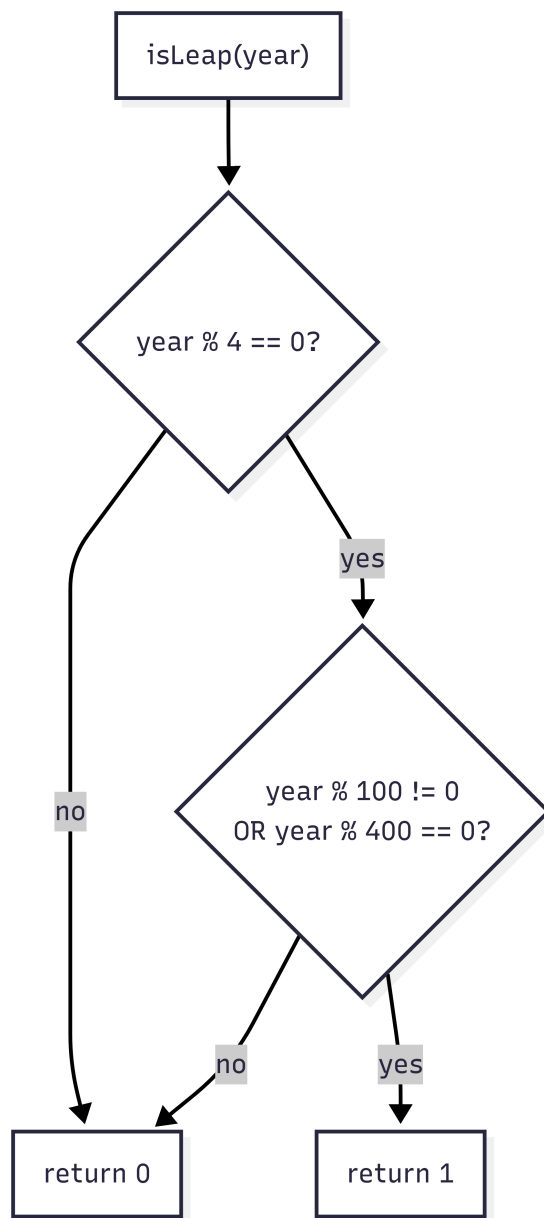


Figure 14: isLeap

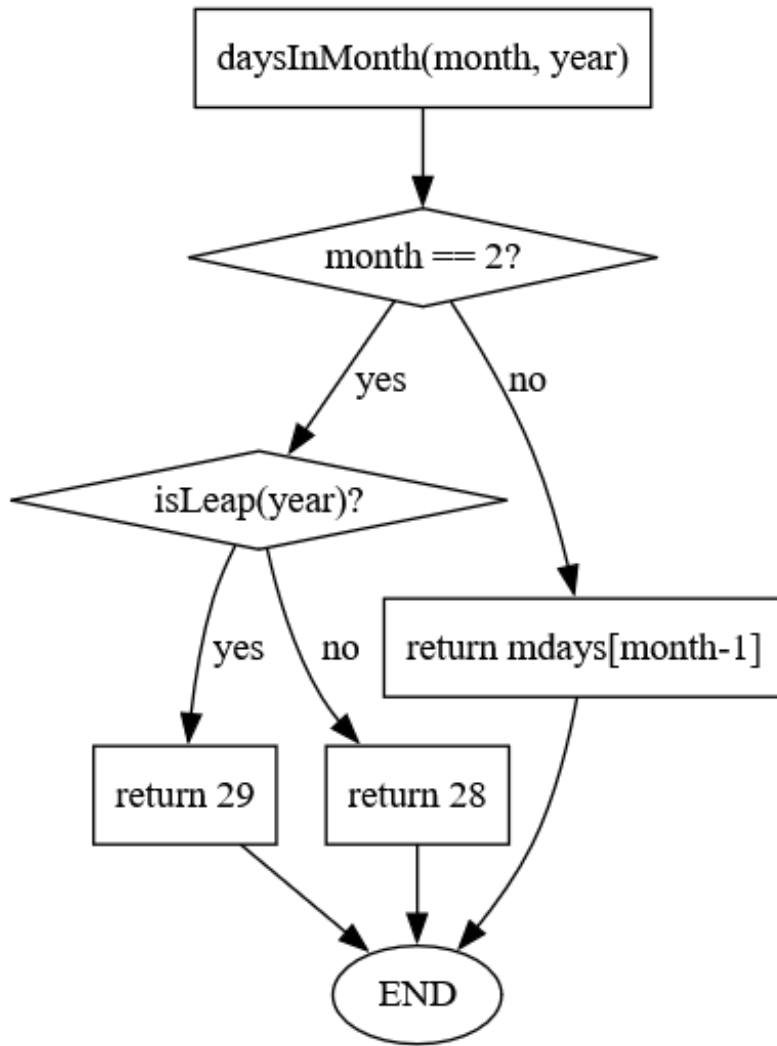


Figure 15: daysInMonth

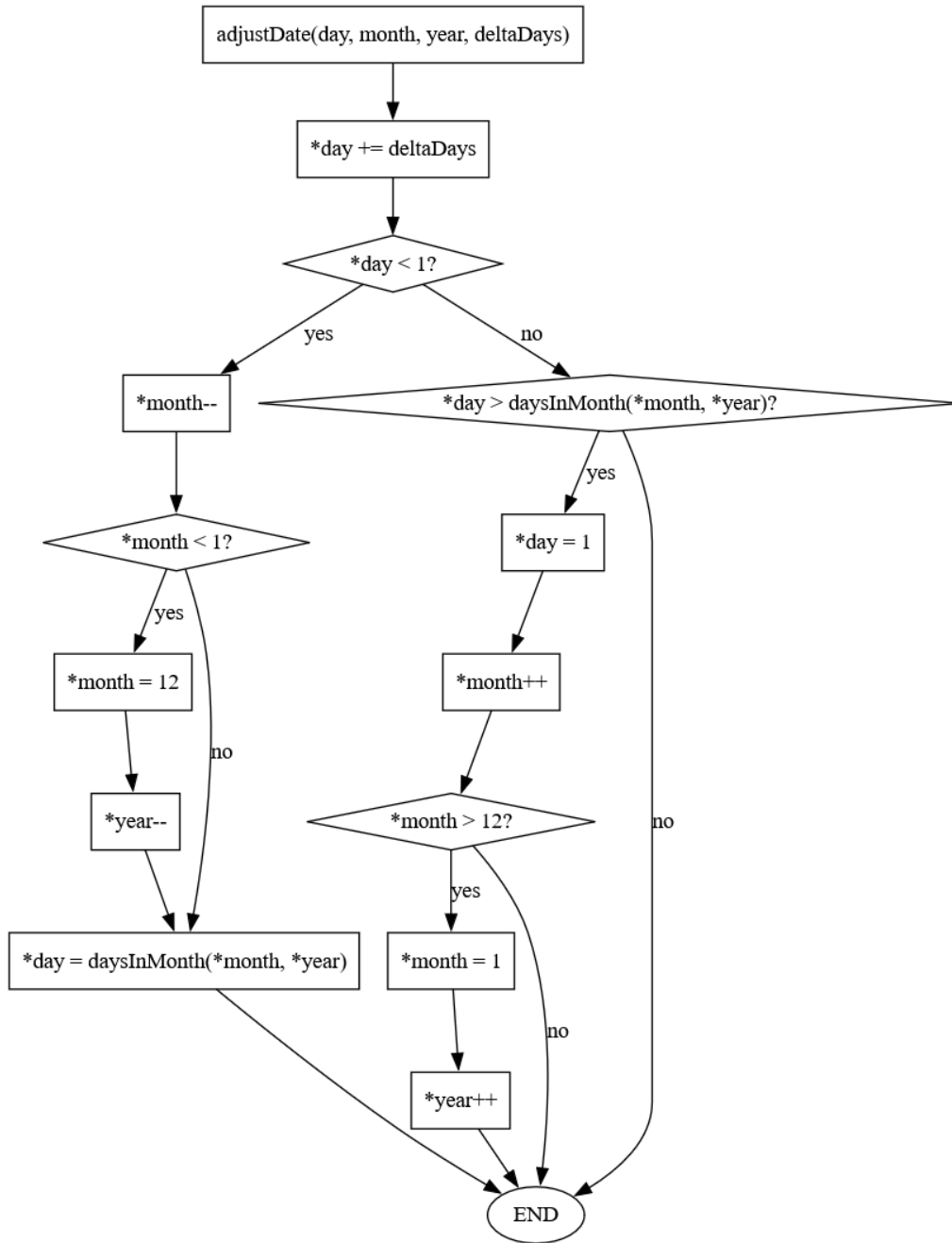


Figure 16: adjustDate

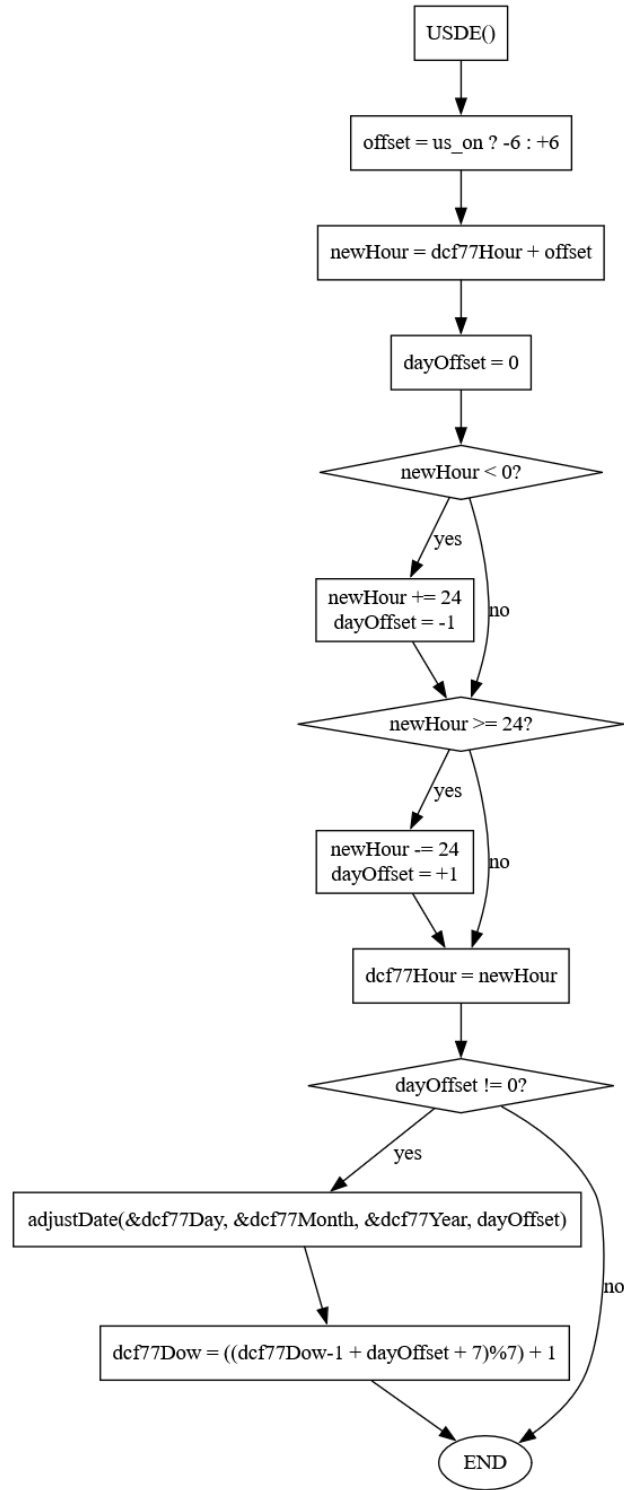


Figure 17: USDE