

```

// -----
//          HUSK RETTE FILNAVN OG DATO i ver 1 og ver 2 herunder
char ver1[ ] = "VFO Si5351B 028B"; // FilNavn til startdisplay
char ver2[ ] = "OZ1LN 27-09-2025"; // OZ1LN+Dato til startdisplay
// -----
// 29-10-2025-028B Forsøg med Si5351B med FM. 9 ports modul. Virker.
// OBS OBS Der er ikke tjek på om syntesen er i lås. Fejlstart er set ;o(

#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
// Display forbundet med D12 til RS, D11 til E, RW til stel.
// D10 til display D4, D9 til D 5, D8 til D6 og D7 til D7.

#include "si5351.h"          // hjælpebibliotek til Si5351 A-B-C
#include "Wire.h"

#define PLLB_FREQ 876000000ULL

Si5351 si5351;

#define XT_CAL_F 139000 // Si5351 Krystal frekvens justering
// Si5351 justeres til at give nøjagtig frekvens.
// Forøgelse vil forminske udgangsfrekvensen og omvendt. 139000
// -----

// deklaration af pins -----

const int KNAP_1 = 6; // D6
const int KNAP_2 = 5; // D5
const int KNAP_3 = 4; // D4
const int KNAP_4 = 3; // D3
const int KNAP_5 = 2; // D2
const int KNAP_6 = 14; // A0

const int blinkled = 13; // D13 pin 13 er LED på NANO

boolean Knap1 = 0;
boolean Knap2 = 0;
boolean Knap3 = 0;
boolean Knap4 = 0;
boolean Knap5 = 0;
boolean Knap6 = 0;

long cal = XT_CAL_F; // bruges til finjustering af Si5351 output frekvens

unsigned long freq_valg = 145500000; // Start frekvens 145,500 MHz
unsigned long freq_mf = 107000000; // mellemfrekvens 107000000 Hz
unsigned long freq_space = 600000; // repeaterspace 00600000 Hz
unsigned long freq_step1 = 12500; // Kanalstep 12,5 kHz
unsigned long freq_step2 = 1000000; // Kanalstep 1 MHz
unsigned long last_freq = freq_valg + 12500; // Display og AD opdateres ved start
unsigned long freq_lcd;
unsigned long freq_dds;
// -----

```

```

// Setup() kører kun 1 gang ved programstart
// -----
void setup()
{
    // DEFINE INPUTS
    pinMode (KNAP_1, INPUT_PULLUP);
    pinMode (KNAP_2, INPUT_PULLUP);
    pinMode (KNAP_3, INPUT_PULLUP);
    pinMode (KNAP_4, INPUT_PULLUP);
    pinMode (KNAP_5, INPUT_PULLUP);
    pinMode (KNAP_6, INPUT_PULLUP);

    lcd.begin(16, 2);           // start 16*2 display
    lcd.setCursor(0, 0);       // øverste linje
    lcd.print(ver1);           // print ver1
    lcd.setCursor(0, 1);       // nederste linje
    lcd.print(ver2);           // ver2

    Serial.begin(9600); // opens serial port, sets data rate to 9600 bps
    Serial.println(ver1);
    Serial.println(ver2);

    delay(2000);               // vent 2 sekunder

    lcd.setCursor(0, 0);       // øverste linje
    lcd.print("                "); // Display ryddes
    lcd.setCursor(0, 1);       // nederste linje
    lcd.print("                "); // Display ryddes

    pinMode(blinkled , OUTPUT); // LED på NANO
    digitalWrite(blinkled , LOW);

    // Tjek om SI5351 er tilstede.
    bool i2c_found;
    i2c_found = si5351.init(SI5351_CRYSTAL_LOAD_8PF, 0, 0);
    if(!i2c_found)
    {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Device not found");
        delay(10000);
    }
    else
    {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Device found");
        delay(1000);
    }
}

```

```

// Set VCXO osc to 876 MHz (146 MHz x 6), 240 ppm pull er MAX
si5351.set_vcxo(PLL_B_FREQ, 240);

// Set CLK0 to be locked to VCXO
si5351.set_ms_source(SI5351_CLK0, SI5351_PLLB);
si5351.set_ms_source(SI5351_CLK1, SI5351_PLLB);

// Tune to 146 MHz center frequency
si5351.set_freq_manual(146000000ULL, PLL_B_FREQ, SI5351_CLK0);
si5351.update_status();
delay(500);
}
// -----Her begynder funktioner-----
void Test_Input()
{
    Knap1 = !digitalRead(KNAP_1); // input inverteres
    Knap2 = !digitalRead(KNAP_2); // input inverteres
    Knap3 = !digitalRead(KNAP_3); // input inverteres
    Knap4 = !digitalRead(KNAP_4); // input inverteres
    Knap5 = !digitalRead(KNAP_5); // input inverteres
    Knap6 = !digitalRead(KNAP_6); // input inverteres

    if (Knap1 == 1)
    {
        freq_valg = freq_valg + freq_step1; // step1 er + 12.5 kHz
        delay(300); // simpel debounce
    }
    if (Knap2 == 1)
    {
        freq_valg = freq_valg - freq_step1; // step1 er - 12.5 kHz
        delay(300); // simpel debounce
    }
    if (Knap3 == 1)
    {
        freq_valg = freq_valg + freq_step2; // step2 er + 1 MHz
        delay(300); // simpel debounce
    }
    if (Knap4 == 1)
    {
        freq_valg = freq_valg - freq_step2; // step2 er - 1 MHz
        delay(300); // simpel debounce
    }
    if (Knap5 == 1)
    {
        freq_valg = 145000000; // sæt frekvens til 145,000 MHz
        delay(300); // simpel debounce
    }
    if (Knap6 == 1)
    {
        freq_valg = 145500000; // sæt frekvens til 145,500 MHz
        delay(300); // simpel debounce
    }
}
// -----

```

```

void format_lcd()

// Frekvensen formateres med 2 decimal punkter, MHz and KHz
{
    char str[12]; // Split frekvens-strengen op, indsæt punkter
    sprintf(str, "%010lu", freq_lcd);
    str[0] = str[1];
    str[1] = str[2];
    str[2] = str[3];
    str[3] = '.';
    str[4] = str[4];
    str[5] = str[5];
    str[6] = str[6];
    str[7] = ' ';
    // str[8] = 'M';
    // str[9] = 'H';
    // str[10] = 'z';
    // str[11] = ' ';

    if (freq_lcd < 10000000) {str[0] = ' '; str[1] = ' ';}
    if (freq_lcd < 100000000) str[0] = ' ';

    lcd.print(str);
}

// -----
// Vis VFO frekvenser i LCD display
// -----
void vis_frekvens()
{
    lcd.clear();

    lcd.setCursor(0, 0);
    lcd.print("RX ");
    freq_lcd = freq_valg;
    format_lcd();

    lcd.setCursor(0, 1);
    lcd.print("VFO ");
    freq_lcd = freq_dds;
    format_lcd();
}
// -----

void dds_5351()
{
    // data freq_dds * 100 sendes til Si5351. CLK0 med FM modulation.
    si5351.set_freq_manual(freq_dds * 100ULL, PLLB_FREQ, SI5351_CLK0);

    // ***** Her sættes Clock1 til 10.7 MHz med FM modulation *****
    si5351.set_freq_manual(10700000 * 100ULL, PLLB_FREQ, SI5351_CLK1);
}

```

```

//*****
// HOVEDPROGRAM. Her står den kode, som kører i uendelig sløjfe.
//*****
void loop()
{
    Test_Input();

    if (freq_valg != last_freq) // Hvis freq er ændret opdateres DDS og Display
    {
        freq_dds = (freq_valg / 3); // Her kommer senere regnestykker for MF, Space etc

        vis_frekvens();           // skriv frekvenser i display

        Serial.println(freq_valg); // skriv frekvens i monitor

        dds_5351();              // send DDS koden til Si5351

        digitalWrite(13, HIGH); // Blink med LED for at vise opdateringen
        delay(50);
        digitalWrite(13, LOW);

        last_freq = freq_valg ;   // sæt last_freq lig med freq_valg
    }
}
// -----Program slut-----

```