#include <SoftwareSerial.h>

SoftwareSerial popeyeza12(10, 11); // RX, TX

void setup() {

  Serial.begin(9600);

  popeyeza12.begin(57600);

  delay(200);  //break for available to read...

}

void loop() {

  delay(100);  //break for available to read...

  //MY COMPUTER

  if(Serial.available()){             //Only Once

    String text1 = "";

    while(Serial.available()){

      char one = Serial.read();

      popeyeza12.write(one);          //READ ME ->SEND

      text1 += one;

      delay(5);  //break time

    }

    Serial.print("Me: ");

    Serial.println(text1);

  }

  //OTHER

  if (popeyeza12.available()){           //TO ME <- RECEIVE

    String text2 = "";

    while(popeyeza12.available()){

      char one = popeyeza12.read();

      text2 += one;

      delay(5);  //break time

    }

    Serial.print("B: ");

    Serial.println(text2);

  }

}

#include <Wire.h> //ASK TX

#include <Adafruit\_MCP4725.h>

Adafruit\_MCP4725 dac;

#define defaultFreq 1700  //DAC speed (Hz)

//freq0 : frequency of carrier sine wave (Hz)

#define freq0 500

//A[0] - A[3] : ASK Amplitude (0,5] (V)

const float A[4] = {1, 2, 3, 4}; // DEFINEE!!!

//S\_DAC: Amplitue(12bit) of sine wave at 0, 90, 180, 270

const float zeta[4] = {0, 90, 180, 270};

float S[4];

uint16\_t S\_DAC[4];

int delay0;

char inData[20]; //Allocate some space for the string

void setup( ) {

  Serial.begin(115200);

  for(int i = 0 ; i < 4 ; i++ ){

    S[i] = sin(zeta[i]/180 \* PI);

  }

  for (int i = 0 ; i < 4 ; i++){

    S\_DAC[i] = map(S[i]\*100,-100,100,0,4095); // STORE S\_DAC!!!!

    Serial.print("S[");

    Serial.print(i);

    Serial.print("] = ");

    Serial.print(S[i]);

    Serial.print(" - S\_DAC[");

    Serial.print(i);

    Serial.print("] = ");

    Serial.println(S\_DAC[i]);

  }

  /\* set buadrate serial is 115200 \*/

  dac.begin(0x64); // set to default for us use regis 0x64

  delay0 = (1000000 / freq0 - 1000000 / defaultFreq) / 4; // same formula

  // delay for sampling period of sine

  // (Tsine – delayfrom DAC processing speed)

  Serial.flush(); // for clear buffer serial

}

void loop( ) {

  if (Serial.available() > 0) { // if have serial input

    // input message

    for (int i = 0; i == 0 || inData[i - 1] != 0xFFFFFFFF; i++) { // first time or before last is -1

      inData[i] = Serial.read(); // Read a character // if read nothing.. will be 0xFFFFFFFF or -1

    }

    // cycle send data 8 bits

    for (int i = 0; inData[i] != 0xFFFFFFFF; i++ ) { // check if read -1

      // LSB to MSB 8 bits

      for (int k = 0; k < 8; k += 2) { // 4 group of 2 bits(LSB to MSB)

        int temp = inData[i] & 3; // 00, 01, 10, 11  only 2 bit right (LSB)

        for (int cycle = 0; cycle < 5; cycle++ ) { // send 5 cycle/baud

          for (int sample = 0; sample < 4; sample++ ) { // gen 4 sample/cycle

            dac.setVoltage(S\_DAC[sample % 4]\*A[temp] / 4, false);

            delayMicroseconds(delay0); // sampling delay

          }

        }

        inData[i] >>= 2;  //Shift 2 bit to the right !!!

      }

    }

    dac.setVoltage(0, false); // default 0 V for don't send

  }

}

#define defaultFreq 1700 //DAC speed (Hz) // ASK RX

//freq0 : frequency of carrier sine wave (Hz)

#define freq0 1700

//A[0]-A[3] : ASK Amplitude (0,5] (V)

const float A[4] = {1, 2, 3, 4};

int delay0;

// amin/amax : Amplitude in digital 10bit

#define a0min 1.13\*1023/5 // a0min <= a0(1v) <= a0max  1.13 - 1.38

#define a0max 1.38\*1023/5

#define a1min 2.26\*1023/5 // a1min <= a1(2v) <= a1max  2.26 - 2.70

#define a1max 2.70\*1023/5

#define a2min 3.45\*1023/5 // a2min <= a2(3v) <= a2max  3.45 - 3.89

#define a2max 3.89\*1023/5

#define a3min 4.71\*1023/5 // a3min <= a3(4) <= a3max   4.71 - 5.00

#define a3max 5.00\*1023/5

#define r\_slope 100

int sum = 0;

int max = 0;

int prev = 0;

int check = false;

int count = 0;

int charCount = 0;

void setup() {

  // set serial baudrate the same as in TX

  Serial.begin(115200);

  Serial.println("Receive Ready!");

}

int text;

void loop() {

  int temp = analogRead(A0); // read signal from analog pin ALWAYS READ!!

  if (temp - prev > r\_slope && check == false) {

    max = 0;

    check = true; // change check status is true

  }

  if (temp > max) { // update max value

    //Serial.println(temp);

    max = temp; // Max amplitude

  }

  if (max - temp > r\_slope) { // check for falling signal

    //Serial.print("Max : ");

    //Serial.print(max);

    //Serial.print(" temp : ");

    //Serial.println(temp);

    if (check == true) {

      int a;

      if (a0min < max && max < a0max) {

        a = 0;

        count++;

        //Serial.print(" 00 ");

      }

      else if (a1min < max && max < a1max) {

        a = 1;

        count++;

        //Serial.print(" 01 ");

      }

      else if (a2min < max && max < a2max) {

        a = 2;

        count++;

        //Serial.print(" 10 ");

      }

      else if (a3min < max && max < a3max) {

        a = 3;

        count++;

        //Serial.print(" 11 ");

      }

      if (count == 5) { // 5 cycles

        int t = (a & 3) << (charCount \* 2); // LAST 2 BIT

        text |= t;    // sum (OR) bit

        charCount++; //

        if (charCount == 4) {

          Serial.print(char(text));

          charCount = 0; // Reset = 0

          text = 0;      // Reset = 0x00

        }

        count = 0;

      }

    }

    check = false; // change check status is false

  }

  prev = temp; // assign temp value to previous

}

#include <Wire.h> // FSK TX

#include <Adafruit\_MCP4725.h>

#define defaultFreq 1700 // dac speed (Hz)

#define baudrate 250

int S\_DAC[4] = {2000,4000,2000,0};  //S\_DAC : Amplitude (12bit) of sine wave at 0,90,180,270

Adafruit\_MCP4725 dac;

int delaySample[4] = {};

int freq[4] = {500,750,1000,1250};

char input[30] = {}; //buffer input

void setup() {

  Serial.begin(115200);

  dac.begin(0x64); // set to default

  for(int i = 0; i < 4; i++){

    delaySample[i] = (1000000 / freq[i] - 1000000 / defaultFreq) / 4; //sampling period for FSK 500,750,1000,1250 Hz

  }

  Serial.println("Enter Text:");

  Serial.flush(); // for clear buffer serial

}

void loop() {

  if (Serial.available() > 0) { // for get input

    int count = 0;

    for (int i = 0 ; Serial.available() > 0; i++) {

      input[i] = Serial.read();

      count++;

    }

    for (int i = 0; i < count; i++) { // Every charactors

      for (int k = 0; k < 4; k++) { //4 Group of 2 bits

        int temp = input[i] & 3; // 00, 01, 10, 11 only 2 bit right

        int useDelay = 0;

        int cyc = 0;

        for (int j = 0; j < 4 ; j++){   // check Encoding Rules (2 bits, 4 samples)

          if (temp == j){

            cyc = freq[j] / baudrate;   // make cycles

            useDelay = delaySample[j];  // make delay

          }

        }

        // SEND !!!

        for (int cycle = 0; cycle < cyc; cycle++) {

          for (int sample = 0; sample < 4; sample++) {

            dac.setVoltage(S\_DAC[sample], false);

            delayMicroseconds(useDelay);

          }

        }

        input[i] >>= 2; // Shift 2 bits to right

      }

      // End 1 Char

    }

    // End every Char

    dac.setVoltage(0, false); // default

  }

}

#define baudrate 250 // FSK RX

int cyc[4] = {500/baudrate, 750/baudrate, 1000/baudrate, 1250/baudrate}; //500,750,1000,1250

//int bits[4] = {0,1,2,3};

// cbi this for increase analogRead Speed

#ifndef cbi

#define cbi(sfr, bit) (\_SFR\_BYTE(sfr) &= ~\_BV(bit))

#endif

#ifndef sbi

#define sbi(sfr, bit) (\_SFR\_BYTE(sfr) |= \_BV(bit))

#endif

// amplitude diff. for detecting rising or falling signal

#define r\_slope 100

void setup() {

  Serial.begin(115200);

  sbi(ADCSRA,ADPS2) ; // this for increase analogRead Speed

  cbi(ADCSRA,ADPS1) ;

  cbi(ADCSRA,ADPS0) ;

  Serial.println("Receive Ready : ");

  Serial.flush();

}

int cycleCount = 0;

unsigned long int now = 0;

unsigned long int past = 0;

int periodTime = 4000;

int prev = 0;

uint16\_t text = 0;

bool check = false;

int max = 0;

int bitCount = 0;

bool lastAmp = true;

void loop() {

  now = micros();

  int temp = analogRead(A0); // read analog wave from A0

  // check period of input analog signal

  if (temp - prev > r\_slope && check == false) { // RISING

    max = 0;

    check = true; // change check status is true

    if (lastAmp == true){

      lastAmp = false;

      past = now; // Reset Time

    }

  }

  if (temp > max) { // update max value  RISING

    max = temp; // Max amplitude

  }

  if (max - temp > r\_slope){ // FALLING

    if (check == true){ // PEAK ALREADY

      cycleCount++;

      check = false;  // calculate input frequency

    }

  }

  if (now - past >= periodTime){

    lastAmp = true;

    for(int i = 0; i < 4; i++){

      if (cyc[i] == cycleCount){  // decode data bits from detected input frequency

         //Serial.print(cycleCount);

         int t = (i & 3) << (bitCount\*2);  //2 bits

         text |= t;

         bitCount++;

         cycleCount = 0;   // Reset cycleCount

      }

    }

    if (bitCount == 4){

      Serial.print(char(text));   // show read data bits

      text = 0;       // Reset text char

      bitCount = 0;   // Reset bitCount

    }

  }

  prev = temp; // assign temp value to previous

}

#include <Wire.h> // FM TX

#include <Adafruit\_MCP4725.h>

#define defaultFreq 1700 // dac speed (Hz)

#define baudrate 250

int S\_DAC[4] = {1000,2000,1000,0};  //S\_DAC : Amplitude (12bit) of sine wave at 0,90,180,270

Adafruit\_MCP4725 dac;

int delaySample[4] = {};

int freq[4] = {500,750,1000,1250};

char input[300] = {}; //buffer input

void setup() {

  Serial.begin(115200);

  dac.begin(0x64); // set to default

  for(int i = 0; i < 4; i++){

    delaySample[i] = (1000000 / freq[i] - 1000000 / defaultFreq) / 4; //sampling period for FSK 500,750,1000,1250 Hz

  }

  dac.setVoltage(0, false); // default

  Serial.println("Enter Text:");

  Serial.flush(); // for clear buffer serial

}

void loop() {

  if (Serial.available() > 0) { // for get input

    int count = 0;

    for (int i = 0 ; Serial.available() > 0; i++) {

      input[i] = Serial.read();

      count++;

    }

    for (int i = 0; i < count; i++) { // Every charactors

      for (int k = 0; k < 4; k++) { //4 Group of 2 bits

        int temp = input[i] & 3; // 00, 01, 10, 11 only 2 bit right

        int useDelay = 0;

        int cyc = 0;

        for (int j = 0; j < 4 ; j++){   // check Encoding Rules (2 bits, 4 samples)

          if (temp == j){

            cyc = freq[j] / baudrate;   // make cycles

            useDelay = delaySample[j];  // make delay

          }

        }

        // SEND !!!

        for (int cycle = 0; cycle < cyc; cycle++) {

          for (int sample = 0; sample < 4; sample++) {

            dac.setVoltage(S\_DAC[sample], false);

            delayMicroseconds(useDelay);

          }

        }

        input[i] >>= 2; // Shift 2 bits to right

      }

      // End 1 Char

    }

    // End every Char

    dac.setVoltage(0, false); // default

  }

}

// TEA5767 Example // FM RX

#include <Wire.h>

#include <TEA5767Radio.h>

TEA5767Radio radio = TEA5767Radio();

#define baudrate 250

int cyc[4] = {500/baudrate, 750/baudrate, 1000/baudrate, 1250/baudrate};

//int bits[4] = {0,1,2,3};

// cbi this for increase analogRead Speed

#ifndef cbi

#define cbi(sfr, bit) (\_SFR\_BYTE(sfr) &= ~\_BV(bit))

#endif

#ifndef sbi

#define sbi(sfr, bit) (\_SFR\_BYTE(sfr) |= \_BV(bit))

#endif

// amplitude diff. for detecting rising or falling signal

#define r\_slope 130// r > 110 130 150 < r //

void setup() {

  Wire.begin();

  radio.setFrequency(107.5); // pick your own frequency

  Serial.begin(115200);

  sbi(ADCSRA,ADPS2) ; // this for increase analogRead Speed

  cbi(ADCSRA,ADPS1) ;

  cbi(ADCSRA,ADPS0) ;

  Serial.println("Receive Ready : ");

  Serial.flush();

}

int cycleCount = 0;

unsigned long int now = 0;

unsigned long int past = 0;

int periodTime = 3700; // 3850

int prev = 450; // middle of voltage

uint16\_t text = 0;

bool check = false;

int min = 800; // min mode

int bitCount = 0;

bool lastAmp = true;

void loop() {

  now = micros();

  int temp = analogRead(A2); // read analog wave from A0

  // check period of input analog signal

  //if (temp > 600) Serial.println(temp);

  if (prev - temp > r\_slope && check == false) { // RISING ## FALLING

    min = 800;

    check = true; // change check status is true

    if (lastAmp == true){

      lastAmp = false;

      past = now; // Reset Time

    }

  }

  if (temp < min) { // update max value  RISING ## FALLING

    min = temp; // Max amplitude ## MIN!

  }

  if (temp - min > r\_slope){ // FALLING ## RISING!!!

    if (check == true){ // PEAK ALREADY

      cycleCount++;

      check = false;  // calculate input frequency

    }

  }

  if (now - past >= periodTime){

    lastAmp = true;

    /\*

    if (cycleCount > 0){

      Serial.print(bitCount);

      Serial.print("  ");

      Serial.println(cycleCount);

      cycleCount = 0;

      bitCount++;

    }

    \*/

    for(int i = 0; i < 4; i++){

      if (cyc[i] == cycleCount){  // decode data bits from detected input frequency

         //Serial.print(cycleCount);

         int t = (i & 3) << (bitCount\*2);  //2 bits

         text |= t;

         bitCount++;

         cycleCount = 0;   // Reset cycleCount

      }

    }

    if (bitCount == 4){

      Serial.print(char(text));   // show read data bits

      //Serial.flush();

      text = 0;       // Reset text char

      bitCount = 0;   // Reset bitCount

    }

  }

  prev = temp; // assign temp value to previous

}