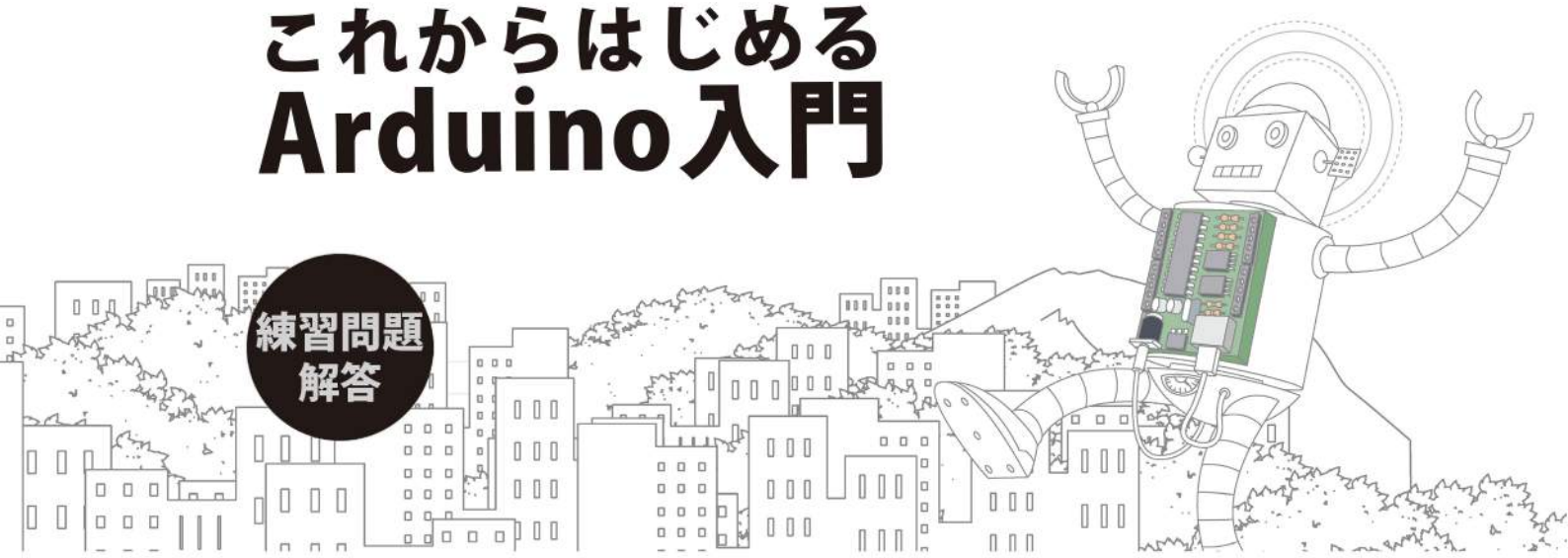
これからはじめる Arduino入門



練習問題 解答

■ 問題 3-1

```
int count = 0;

void setup(){
  Serial.begin(9600);
}

void loop(){
  count++;
  Serial.println("Hello World !! " + String(count));
  delay(1000);
}
```

■ 問題 5-1

```
int ledOut = 8;
int buttonIn = 9;

void setup() {
  pinMode(ledOut, OUTPUT);
  pinMode(buttonIn, INPUT);
}

void loop() {
  int buttonState = digitalRead(buttonIn);
  if(buttonState == LOW)
    digitalWrite(ledOut, HIGH);
  else
    digitalWrite(ledOut, LOW);
}
```

■ 問題 5-2

```
int LED_pin = 3;

void setup()
{
  pinMode(LED_pin, OUTPUT);
}

void loop()
{
  int i;

  for(i = 0; i < 256; i++){
```

```
    analogWrite(LED_pin, i);
    delay(10);
  }
  for(i = 255; i >= 0; i--){
    analogWrite(LED_pin, i);
    delay(10);
  }
}
```

■ 問題 6-1

```
void setup(){
  Serial.begin(9600);
}

void loop(){
  byte num = 0b01001001;

  Serial.println("Encoded value : " + String(num));
  Serial.println("Decimal       : " + String(convert2decimal(num)));

  while(true);
}

byte convert2decimal(byte num){
  byte low = num & 0x0F;
  byte high = (num & 0xF0) >> 4;

  return high * 10 + low;
}
```

■ 問題 6-2

```
void setup(){
  Serial.begin(9600);
}

void rectangular2polar(int x, int y, float *rho, float *theta){
  *rho = sqrt(x * x + y * y);
  *theta = int(atan(float(y) / x) * 180 / 3.14);
}

void loop(){
  int x = 1, y = 1;
  float rho, theta;

  rectangular2polar(x, y, &rho, &theta);

  Serial.print("(" + String(x) + ", " + String(y) + ") => (");
  Serial.print(rho);
  Serial.print(", ");
  Serial.print(theta);
  Serial.println(")");

  while(true);
}
```

■ 問題 6-3

```
void setup(){
  Serial.begin(9600);
}

void loop(){
  Serial.println("With while    : " + String(sum_with_while(100)));
  Serial.println("With do-while : " + String(sum_with_do_while(100)));
  Serial.println("With for      : " + String(sum_with_for(100)));

  while(true);
}
```

```
int sum_with_while(int n){
  int sum = 0, count = 0;

  while(count <= n){
    if(count % 2 == 0 && count % 3 != 0)
      sum += count;
    count++;
  }

  return sum;
}

int sum_with_do_while(int n){
  int sum = 0, count = 0;

  do{
    if(count % 2 == 0 && count % 3 != 0)
      sum += count;
    count++;
  }while(count <= n);

  return sum;
}

int sum_with_for(int n){
  int sum = 0, count;

  for(count = 0; count <= n; count++){
    if(count % 2 == 0 && count % 3 != 0)
      sum += count;
  }

  return sum;
}
```

■ 問題 6-4

```
int count = 0;
unsigned long time_previous, time_current;

void setup(){
  Serial.begin(9600);
  time_previous = millis();
}

void loop(){
  time_current = millis();

  if(time_current - time_previous >= 1000){
    time_previous = time_current;
    count++;
    Serial.println(String("Count : ") + count);
  }
}
```

■ 問題 6-5

```
int count1 = 0, count2 = 0;
unsigned long time_previous1, time_previous2, time_current;

void setup(){
  Serial.begin(9600);
  time_previous1 = millis();
  time_previous2 = millis();
}

void loop(){
  time_current = millis();

  if(time_current - time_previous1 >= 1000){
    time_previous1 = time_current;
    count1++;
    Serial.println(String("Count1 : ") + count1);
  }
```

```
  if(time_current - time_previous2 >= 2000){
    time_previous2 = time_current;
    count2++;
    Serial.println(String("Count2 : ") + count2);
  }
}
```

■ 問題 7-1

```
int ledPin = 3;
int analogPin = A0;
int val = 0;

void setup()
{
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  val = analogRead(analogPin);
  int mapped = 255 - 7 * sqrt(val);
  Serial.print(String(val) + " : " + mapped);
  analogWrite(ledPin, mapped);
}
```

■ 問題 7-2

```
void setup(){
  Serial.begin(9600);
}

void loop(){
  Serial.println("Degree\tSin\tCos\tTan");
  Serial.println("============================");
  for(int i = 0; i <= 80; i += 10){
    float rad = i * 3.14 / 180.0;
    Serial.print(i);
    Serial.print("\t");
    Serial.print(sin(rad));
    Serial.print("\t");
    Serial.print(cos(rad));
    Serial.print("\t");
    Serial.println(tan(rad));
  }
  while(true);
}
```

■ 問題 7-3

```
void setup(){
  Serial.begin(9600);
  randomSeed(analogRead(A0));
}

void loop(){
  int count[10] = {0,};

  for(int i = 0; i < 1000; i++){
    int val = random(10);
    count[val]++;
  }

  for(int i = 0; i < 10; i++){
    Serial.println("Value " + String(i) + " : " + String(count[i]));
  }

  while(true);
}
```

問題 8-1

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int count = 0;
  String str = "123  45  23";
  int startIndex = 0, endIndex, n = str.length();

  do{
    endIndex = str.indexOf(' ', startIndex);

    if(endIndex == -1 && startIndex < n - 1) endIndex = n;

    String subStr = str.substring(startIndex, endIndex);
    subStr.trim();
    if(subStr.length() > 0){
      count++;
      Serial.print(String(count) + "th number : ");
      Serial.println(subStr);
    }
    startIndex = endIndex + 1;
  }while(startIndex < n - 1);

  while(true);
}
```

問題 8-2

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int count = 0;
  char str[] = "  123, 45  23  ";
  int startIndex = 0, endIndex, n = strlen(str);
  String subStr;

  for(int i = 1; i < n; i++){
    if(str[i] == ' ' || str[i] == ',' || i == n - 1){
      if(i == n - 1)
        subStr = String(str).substring(startIndex);
      else
        subStr = String(str).substring(startIndex, i);
      subStr.trim();
      if(subStr.length() > 0){
        count++;
        Serial.print(String(count) + "th number : ");
        Serial.println(subStr);
      }
      startIndex = i + 1;
    }
  }

  while(true);
}
```

問題 9-1

```
//
// BigSerial2.h
//
#ifndef BIG_SERIAL2
#define BIG_SERIAL2

#include "Arduino.h"

static int patterns[10][5] = {  // 0から9までの数字パターン
  {0x1F, 0x11, 0x11, 0x11, 0x1F},
  {0x01, 0x01, 0x01, 0x01, 0x01},
```

```
  {0x1F, 0x01, 0x1F, 0x10, 0x1F},
  {0x1F, 0x01, 0x1F, 0x01, 0x1F},
  {0x11, 0x11, 0x1F, 0x01, 0x01},
  {0x1F, 0x10, 0x1F, 0x01, 0x1F},
  {0x10, 0x10, 0x1F, 0x11, 0x1F},
  {0x1F, 0x01, 0x01, 0x01, 0x01},
  {0x1F, 0x11, 0x1F, 0x11, 0x1F},
  {0x1F, 0x11, 0x1F, 0x01, 0x01}};

void printSerialBigNumber(int no);

#endif

//
// BigSerial2.cpp
//
#include "Arduino.h"
#include "BigSerial2.h"

void printSerialBigNumber(int no){
  String numString = String(no);

  for(int i = 0; i < 5; i++){
    for(int index = 0; index < numString.length(); index++){
      int digit = numString[index] - '0';
      char oneLine = patterns[digit][i];
      for(int j = 4; j >= 0; j--){
        char oneBit = (oneLine & (0x01 << j)) >> j;
        if(oneBit == 1)
          Serial.print('O');
        else
          Serial.print('.');
      }
      Serial.print(' ');
    }
    Serial.println();
  }
}
```

■ 問題 11-1

```
int pins[] = {9, 10, 11, 12};
int mode = 1;
int state = 0;

void setup(){
  for(int i = 0; i < 4; i++){
    pinMode(pins[i], OUTPUT);
  }

  Serial.begin(9600);
}

void loop(){
  if(Serial.available()){
    char ch = Serial.read();

    if(ch == '1'){
      mode = 1;
      state = 0;
    }
    else if(ch == '2'){
      mode = 2;
      state = 0;
    }
  }

  Serial.println("Mode " + String(mode) + " State " + String(state));
  if(mode == 1){
    for(int i = 0; i < 4; i++){
      if(i % 2 == state)
        digitalWrite(pins[i], HIGH);
      else
        digitalWrite(pins[i], LOW);
    }
```

```
    state = (state + 1) % 2;
  }
  else if(mode == 2){
    for(int i = 0; i < 4; i++){
      if(i == state)
        digitalWrite(pins[i], HIGH);
      else
      digitalWrite(pins[i], LOW);
    }
    state = (state + 1) % 4;
  }

  delay(500);
}
```

■ 問題 11-2

```
int pins[] = {9, 10, 11, 12};
int mode = 1;
int state = 0;

long previous, current;

void setup(){
  for(int i = 0; i < 4; i++){
    pinMode(pins[i], OUTPUT);
  }

  Serial.begin(9600);
  previous = millis();
}

void loop(){
  if(Serial.available()){
    char ch = Serial.read();

    if(ch == '1'){
      mode = 1;
      state = 0;
    }
    else if(ch == '2'){
      mode = 2;
      state = 0;
    }
  }

  current = millis();
  if(current - previous >= 500)
  {
    Serial.println("Mode " + String(mode) + " State " + String(state));
    previous = current;
    if(mode == 1){
      for(int i = 0; i < 4; i++){
        if(i % 2 == state)
          digitalWrite(pins[i], HIGH);
        else
          digitalWrite(pins[i], LOW);
      }
      state = (state + 1) % 2;
    }
    else if(mode == 2){
      for(int i = 0; i < 4; i++){
        if(i == state)
          digitalWrite(pins[i], HIGH);
        else
        digitalWrite(pins[i], LOW);
      }
      state = (state + 1) % 4;
    }
  }
}
```

問題 12-1

```
#include <SoftwareSerial.h>

int RX = 10;
int TX = 11;

SoftwareSerial mySerial = SoftwareSerial(RX, TX);

void setup(){
  mySerial.begin(9600);
  Serial.begin(9600);
}

void loop(){
  if(mySerial.available()){
    int incoming = mySerial.read();
    Serial.write(incoming);
    mySerial.write(incoming);
  }
}
```

問題 13-1

```
unsigned long previous, current;
boolean led_state = false;

void setup(){
  pinMode(9, OUTPUT);
  pinMode(10, INPUT);
  previous = millis();
}

void loop(){
  boolean button_state = digitalRead(10);

  if(button_state){
    digitalWrite(9, HIGH);
  }
  else{
    current = millis();
    if(current - previous >= 1000){
      previous = current;
      led_state = !led_state;
      digitalWrite(9, led_state);
    }
  }
}
```

問題 13-2

```
unsigned long previous, current;
boolean led_state = false;

void setup(){
  pinMode(9, OUTPUT);
  previous = millis();
}

void loop(){
  int val = analogRead(A0);
  int interval = map(val, 0, 1023, 100, 1000);

  current = millis();
  if(current - previous >= interval){
    previous = current;
    led_state = !led_state;
    digitalWrite(9, led_state);
  }
}
```

問題 14-1

```
int pins[] = {6, 7, 8, 9, 10};

void setup(){
  for(int i = 0; i < 5; i++)
    pinMode(pins[i], OUTPUT);
}

void loop(){
  int value = analogRead(A0);
  // 照度範囲は任意に決定したため実験に合わせて調整しなければならない
  int count = map(value, 400, 900, 1, 5);

  for(int i = 1; i <= count; i++)
    digitalWrite(pins[i - 1], HIGH);

  for(int i = count + 1; i <= 5; i++)
    digitalWrite(pins[i - 1], LOW);

  delay(100);
}
```

問題 14-2

```
int pins[] = {6, 7, 8, 9, 10};    // LED

void setup() {
  for(int i = 0; i < 5; i++)
    pinMode(pins[i], OUTPUT);
}

void loop() {
  int light = analogRead(A2);     // 照度センサ
  delay(10);
  light = analogRead(A2);
  delay(10);

  int val = analogRead(A1);       // 温度センサ
  delay(10);
  val = analogRead(A1);
  delay(10);
  float voltage = val * 5.0 / 1024.0;
  float temperature = (voltage - 0.5) * 100;

  // 照度と温度範囲は任意に決定したため実験に合わせて調整しなければならない
  int count1 = map(light, 400, 900, 1, 5);
  int count2 = map(temperature, 20, 40, 1, 5);
  int count = max(count1, count2);

  for(int i = 1; i <= count; i++)
    digitalWrite(pins[i - 1], HIGH);

  for(int i = count + 1; i <= 5; i++)
    digitalWrite(pins[i - 1], LOW);

  delay(500);
}
```

問題 15-1

```
int max_distance = 60;
int distance;

int triggerPin = 7;
int echoPin = 6;
int ledPin = 3;

void setup() {
  pinMode(triggerPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(ledPin, OUTPUT);
}
```

```
void loop()
{
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);

  distance = pulseIn(echoPin, HIGH) / 58;  // in cm

  int brightness;
  if(distance > max_distance) brightness = 0;
  else brightness = map(distance, 0, max_distance, 255, 0);

  analogWrite(ledPin, brightness);

  delay(500);
}
```

■ 問題 15-2

```
#include <pitches.h>

int speakerPin = 13;
int notes[] = {0, NOTE_C4, NOTE_D4, NOTE_E4, NOTE_F4,
               NOTE_G4, NOTE_A4, NOTE_B4, NOTE_C5};
int max_distance = 60;
int min_distance = 10;
int distance;

int triggerPin = 7;
int echoPin = 6;

void setup() {
  pinMode(triggerPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(speakerPin, OUTPUT);
}

void loop()
{
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);

  distance = pulseIn(echoPin, HIGH) / 58;  // in cm

  int toneIndex;
  if(distance > max_distance) toneIndex = 0;
  else if(distance < min_distance) toneIndex = 0;
  else toneIndex = map(distance, min_distance, max_distance, 1, 8);

  tone(speakerPin, notes[toneIndex], 30);
}
```

■ 問題 16-1

```
int ON = LOW;
int OFF = HIGH;

int digits[6][8] = {
  {ON, OFF, OFF, OFF, OFF, OFF, OFF, OFF},
  {OFF, ON, OFF, OFF, OFF, OFF, OFF, OFF},
  {OFF, OFF, ON, OFF, OFF, OFF, OFF, OFF},
  {OFF, OFF, OFF, ON, OFF, OFF, OFF, OFF},
  {OFF, OFF, OFF, OFF, ON, OFF, OFF, OFF},
  {OFF, OFF, OFF, OFF, OFF, ON, OFF, OFF}
};

int pins[] = {2, 3, 4, 5, 6, 7, 8, 9};

void setup() {
  for(int i = 0; i < 8; i++){
    pinMode(pins[i], OUTPUT);
  }
```

```
}

void loop() {
  for(int i = 0; i < 6; i++){
    for(int j = 0; j < 8; j++){
      digitalWrite(pins[j], digits[i][j]);
    }
    delay(1000);
  }
}
```

■ 問題 16-2

```
int ON = LOW;
int OFF = HIGH;
int buttonPin = 10;

int count = 0;
int difference = 1;

unsigned long previous, current;

int digits[10][8] = {
  {ON, ON, ON, ON, ON, ON, OFF, OFF},
  {OFF, ON, ON, OFF, OFF, OFF, OFF, OFF},
  {ON, ON, OFF, ON, ON, OFF, ON, OFF},
  {ON, ON, ON, ON, OFF, OFF, ON, OFF},
  {OFF, ON, ON, OFF, OFF, ON, ON, OFF},
  {ON, OFF, ON, ON, OFF, ON, ON, OFF},
  {ON, OFF, ON, ON, ON, ON, ON, OFF},
  {ON, ON, ON, OFF, OFF, ON, OFF, OFF},
  {ON, ON, ON, ON, ON, ON, ON, OFF},
  {ON, ON, ON, OFF, OFF, ON, ON, OFF}};

int pins[] = {2, 3, 4, 5, 6, 7, 8, 9};

void setup() {
  for(int i = 0; i < 8; i++){
    pinMode(pins[i], OUTPUT);
  }
  pinMode(buttonPin, INPUT);
  previous = millis();
}

void loop() {
  current = millis();

  if(digitalRead(buttonPin)){
    difference *= -1;
    delay(200);
  }

  if(current - previous >= 1000){
    previous = current;
    count += difference;
    if(count >= 10) count = 0;
    if(count < 0) count = 9;

    for(int j = 0; j < 8; j++){
      digitalWrite(pins[j], digits[count][j]);
    }
  }
}
```

■ 問題 17-1

```
int digit_select_pin[] = {1, 2, 3, 4};
int segment_pin[] = {5, 6, 7, 8, 9, 10, 11, 12};

int ON = HIGH;
int OFF = LOW;

int digits[6][8] = {
  {ON, OFF, OFF, OFF, OFF, OFF, OFF, OFF},
  {OFF, ON, OFF, OFF, OFF, OFF, OFF, OFF},
```

```
  {OFF, OFF, ON, OFF, OFF, OFF, OFF, OFF},
  {OFF, OFF, OFF, ON, OFF, OFF, OFF, OFF},
  {OFF, OFF, OFF, OFF, ON, OFF, OFF, OFF},
  {OFF, OFF, OFF, OFF, OFF, ON, OFF, OFF}
};

void setup(){
  for(int i = 0; i < 4; i++){
    pinMode(digit_select_pin[i], OUTPUT);
  }
  for(int i = 0; i < 8; i++){
    pinMode(segment_pin[i], OUTPUT);
  }
}

void loop(){
  for(int i = 0; i < 4; i++){
    for(int k = 0; k < 4; k++){
      if(i == k) digitalWrite(digit_select_pin[k], LOW);
      else digitalWrite(digit_select_pin[k], HIGH);
    }

    for(int j = 0; j < 6; j++){
      for(int k = 0; k < 8; k++){
        digitalWrite(segment_pin[k], digits[j][k]);
      }
      delay(200);
    }
  }
}
```

■ 問題 17-2

```
int digit_select_pin[] = {1, 2, 3, 4};
int segment_pin[] = {5, 6, 7, 8, 9, 10, 11, 12};
int time_delay = 5;
boolean going = true;

int minutes = 12;
int seconds = 23;
unsigned long previous_time, current_time;

byte digits_data[10] = {0xFC, 0x60, 0xDA, 0xF2, 0x66, 0xB6, 0xBE, 0xE4, 0xFE, 0xE6};

void setup(){
  for(int i = 0; i < 4; i++){
    pinMode(digit_select_pin[i], OUTPUT);
  }
  for(int i = 0; i < 8; i++){
    pinMode(segment_pin[i], OUTPUT);
  }
  previous_time = millis();
  pinMode(A0, INPUT);
}

void show_digit(int pos, int number){
  for(int i = 0; i < 4; i++){
    if(i + 1 == pos)
      digitalWrite(digit_select_pin[i], LOW);
    else
      digitalWrite(digit_select_pin[i], HIGH);
  }

  for(int i = 0; i < 8; i++){
    byte segment_data = (digits_data[number] & (0x01 << i)) >> i;
    if(segment_data == 1)
      digitalWrite(segment_pin[7 - i], HIGH);
    else
      digitalWrite(segment_pin[7 - i], LOW);
  }
}

void loop(){
  if(digitalRead(A0) == HIGH){
    going = !going;
```

```
    delay(200);
  }

  current_time = millis();
  if(going){
    unsigned long delta = current_time - previous_time;
    if(delta >= 1000L){
      previous_time = current_time;
      seconds++;
      if(seconds == 60){
        seconds = 0;
        minutes++;
        if(minutes == 60) minutes = 0;
      }
    }
  }
  else{
    previous_time = current_time;
  }

  delay(time_delay);
  show_digit(1, minutes / 10);
  delay(time_delay);
  show_digit(2, minutes % 10);
  delay(time_delay);
  show_digit(3, seconds / 10);
  delay(time_delay);
  show_digit(4, seconds % 10);
  delay(time_delay);
}
```

■ 問題 18-1

```
int rows[] = {2, 3, 4, 5, 6, 7, 8, 9};
int cols[] = {10, 11, 12, 13, A0, A1, A2, A3};

byte smile_char[] = {
  B00111100,
  B01000010,
  B10100101,
  B10000001,
  B10100101,
  B10011001,
  B01000010,
  B00111100 };

unsigned long previous, current;
int start_row = 0;

void setup(){
  for(int i = 0; i < 8; i++){
    pinMode(rows[i], OUTPUT);
    pinMode(cols[i], OUTPUT);
  }
  previous = millis();
}

void clear(){
  for(int i = 0; i < 8; i++){
    digitalWrite(rows[i], LOW);
    digitalWrite(cols[i], HIGH);
  }
}

void loop(){
  current = millis();
  if(current - previous > 500){
    previous = current;
    start_row = (start_row + 1) % 16;
  }

  for(int row = 0; row < 8; row++){
    clear();
    digitalWrite(rows[row], HIGH);
    int index = (row + start_row) % 16;
```

```
    char pattern;
    if(index > 7) pattern = 0;
    else pattern = smile_char[index];

    for(int col = 0; col < 8; col++){
      int ox = bitRead(pattern, col);
      if(ox == 1)
        digitalWrite(cols[col], LOW);
      else
        digitalWrite(cols[col], HIGH);
    }
    delay(2);
  }
}
```

■ **問題18–2**

```
int rows[] = {2, 3, 4, 5, 6, 7, 8, 9};
int cols[] = {10, 11, 12, 13, A0, A1, A2, A3};

byte no[][8] = {
  { B00111100, B01000010, B01000010, B01000010, B01000010, B01000010, B01000010, B00111100 },
  { B00001000, B00011000, B00001000, B00001000, B00001000, B00001000, B00001000, B00001000 },
  { B00111100, B01000010, B00000010, B00000100, B00001000, B00010000, B00100000, B01111110 },
  { B00111100, B01000010, B00000010, B00011100, B00000010, B00000010, B01000010, B00111100 },
  { B00001100, B00010100, B00100100, B01000100, B01111110, B00000100, B00000100, B00000100 },
  { B01111110, B01000000, B01000000, B01111100, B00000010, B00000010, B00000010, B01111100 },
  { B00111100, B01000010, B01000000, B01111100, B01000010, B01000010, B01000010, B00111100 },
  { B01111110, B00000010, B00000010, B00000100, B00001000, B00010000, B00100000, B01000000 },
  { B00111100, B01000010, B01000010, B00111100, B01000010, B01000010, B01000010, B00111100 },
  { B00111100, B01000010, B01000010, B00111100, B00000100, B00001000, B00010000, B00100000 }};

unsigned long previous, current;
int index = 0;

void setup(){
  for(int i = 0; i < 8; i++){
    pinMode(rows[i], OUTPUT);
    pinMode(cols[i], OUTPUT);
  }
  previous = millis();
}

void clear(){
  for(int i = 0; i < 8; i++){
    digitalWrite(rows[i], LOW);
    digitalWrite(cols[i], HIGH);
  }
}

void loop(){
  current = millis();
  if(current - previous >= 1000){
    previous = current;
    index = (index + 1) % 10;
  }

  for(int row = 0; row < 8; row++){
    clear();
    digitalWrite(rows[row], HIGH);

    char pattern = no[index][row];
    for(int col = 0; col < 8; col++){
      int ox = bitRead(pattern, col);
      if(ox == 1)
        digitalWrite(cols[7 - col], LOW);
      else
        digitalWrite(cols[7 - col], HIGH);
    }
    delay(2);
  }
}
```

問題19–1

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // RSピン、Eピン、データピン4本

void setup() {
  lcd.begin(16, 2);                    // LCDサイズ指定、2行16文字
  randomSeed(analogRead(A0));
  lcd.clear();
}

void loop() {
  int x = random(0, 17);
  int y = random(0, 2);
  char ch = random('A', 'Z' + 1);

  lcd.setCursor(x, y);
  lcd.write(ch);

  delay(1000);
}
```

問題19–2

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  lcd.begin(16, 2);
  lcd.print("ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890!@#$");
  lcd.print("ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890!@#$");
}

void loop() {
  for(int i = 0; i < 10; i++){
    lcd.scrollDisplayRight();
    delay(500);
  }

  for(int i = 0; i < 10; i++){
    lcd.scrollDisplayLeft();
    delay(500);
  }
}
```

問題20–1

```
#include <glcd.h>
#include <glcd_Buildinfo.h>
#include <glcd_Config.h>
#include <fonts/allFonts.h>

int BUTTON_PIN = 12;

void setup(){
  GLCD.Init();

  GLCD.SelectFont(System5x7);
  pinMode(BUTTON_PIN, INPUT);
}

void loop(){
  int value = digitalRead(BUTTON_PIN);
  GLCD.ClearScreen();

  if(value == HIGH){
    GLCD.print("Button is ON!");
  }
  else{
    GLCD.print("Button is OFF!");
  }

  delay(500);
}
```

問題20–2

```
#include <glcd.h>
#include <glcd_Buildinfo.h>
#include <glcd_Config.h>
#include <fonts/allFonts.h>

int BUTTON_PIN = 12;

void setup(){
  GLCD.Init();

  GLCD.SelectFont(System5x7);
  pinMode(BUTTON_PIN, INPUT);
}

void loop(){
  int value = digitalRead(BUTTON_PIN);
  GLCD.CursorTo(0, 0);

  if(value == HIGH){
    GLCD.print("Button is ON!");
  }
  else{
    GLCD.print("Button is OFF!");
  }
}
```

問題21–1

```
#include <SPI.h>
#include <TFT.h>

#define CS    10
#define DC    9
#define RST   8

TFT screen = TFT(CS, DC, RST);

void setup() {
  screen.begin();
  screen.background(255, 255, 255);
  screen.stroke(0, 0, 0);
}

void loop() {
  int value = analogRead(A0);

  String message = String("Current value : ") + value;
  char buffer[31];

  message.toCharArray(buffer, 30);
  screen.text(buffer, 10, 10);

  delay(100);
}
```

問題21–2

```
#include <SPI.h>
#include <TFT.h>

#define CS    10
#define DC    9
#define RST   8

TFT screen = TFT(CS, DC, RST);

void setup() {
  screen.begin();
  screen.background(255, 255, 255);
}
```

```
void loop() {
  int value = analogRead(A0);

  String message = String("Current value : ") + value;
  char buffer[31];

  message.toCharArray(buffer, 30);

  screen.stroke(255, 255, 255);
  screen.fill(255, 0, 0);    // for demonstration
  screen.rect(105, 9, 25, 9);

  screen.stroke(0, 0, 0);
  screen.text(buffer, 10, 10);

  delay(100);
}
```

■ 問題22-1

```
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2, 3);

String buffer = "";
int LED_pin = 13;
boolean message_end = false;

void setup()
{
  Serial.begin(9600);
  BTSerial.begin(9600);
  pinMode(LED_pin, OUTPUT);
}

void loop()
{
  while(BTSerial.available()){
    char data = BTSerial.read();
    if(data == '\n' || data == '\r'){
      message_end = true;
      break;
    }
    else{
      buffer = buffer + data;
    }
  }

  if(message_end){
    message_end = false;

    if(buffer.equals("on"))
      digitalWrite(LED_pin, HIGH);
    else if(buffer.equals("off"))
      digitalWrite(LED_pin, LOW);

    Serial.println(buffer);
    buffer = "";
  }
}
```

問題22–2

```
// Master
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2, 3);

int LED_pin = 13;
boolean state = false;

void setup()
{
  Serial.begin(9600);
  BTSerial.begin(9600);
  pinMode(LED_pin, OUTPUT);
}

void loop()
{
  if(BTSerial.available()){
    char ch = BTSerial.read();
    if(ch == '1') state = !state;
    digitalWrite(LED_pin, state);
  }
}

// Slave
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2, 3);

int button_pin = 7;

void setup()
{
  Serial.begin(9600);
  BTSerial.begin(9600);    // Bluetoothモジュールの通信速度のデフォルト値
  pinMode(button_pin, INPUT);
}

void loop()
{
  boolean bt = digitalRead(button_pin);
  if(bt){
    Serial.println(bt);
    BTSerial.print(bt);

    delay(200);
  }
}
```

問題23–2

```
// channel A
int DIRECTION_PIN = 12;
int VELOCITY_PIN = 3;
int BRAKE_PIN = 9;

void setup() {
  pinMode(DIRECTION_PIN, OUTPUT);
  pinMode(BRAKE_PIN, OUTPUT);
  digitalWrite(BRAKE_PIN, LOW);
}

void loop(){
  int value = analogRead(A2);
  value = map(value, 0, 1023, -255, 255);

  if(value > 0)
    digitalWrite(DIRECTION_PIN, HIGH);
  else
    digitalWrite(DIRECTION_PIN, LOW);

  analogWrite(VELOCITY_PIN, abs(value));

  delay(100);
}
```

問題24–1

```
#include <Wire.h>
#include "RTClib.h"

RTC_DS1307 RTC;
int button_pin = 2;

void setup () {
  Serial.begin(9600);
  Wire.begin();
  RTC.begin();
}

void loop () {
  if(digitalRead(button_pin)){
    DateTime now = RTC.now();
    Serial.print(now.year(), DEC);
    Serial.print('-');
    Serial.print(now.month(), DEC);
    Serial.print('-');
    Serial.print(now.day(), DEC);
    Serial.print(' ');

    Serial.print(now.hour(), DEC);
    Serial.print(':');
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    Serial.print(now.second(), DEC);
    Serial.println();

    delay(100);
  }
}
```

問題24–2

```
#include <LiquidCrystal.h>
#include <Wire.h>
#include "RTClib.h"

LiquidCrystal lcd(2, 3, 5, 6, 7, 8);
RTC_DS1307 RTC;

void setup(){
  lcd.begin(16, 2);
  Wire.begin();
  RTC.begin();
}

void loop(){
  char buffer1[11], buffer2[9];
  DateTime now = RTC.now();

  sprintf(buffer1, "%04d-%02d-%02d", now.year(), now.month(), now.day());
  sprintf(buffer2, "%02d:%02d:%02d", now.hour(), now.minute(), now.second());

  lcd.setCursor(0, 0);
  lcd.print(buffer1);
  lcd.setCursor(0, 1);
  lcd.print(buffer2);

  delay(1000);
}
```

問題25–1

```
#include <Wire.h>
#define RTC_ADDRESS 0x68

boolean printTime = false;

byte bin2bcd(int n){
```

```
  byte val = 0;

  int ten = n / 10;
  int one = n % 10;

  val = (ten << 4) | one;

  return val;
}

byte bcd2bin(byte n){
  byte val = 0;

  int ten = n >> 4;
  int one = n & 0x0F;

  val = ten * 10 + one;

  return val;
}

void timeSetting(int _year, int _month, int _day, int _hour, int _min, int _sec){
  Wire.beginTransmission(RTC_ADDRESS);
  Wire.write(0);
  Wire.write(bin2bcd(_sec));
  Wire.write(bin2bcd(_min));
  Wire.write(bin2bcd(_hour));
  Wire.write(0);
  Wire.write(bin2bcd(_day));
  Wire.write(bin2bcd(_month));
  Wire.write(bin2bcd(_year - 2000));

  Wire.write(0x10);
  Wire.endTransmission();
}

void printCurrentTime(){
  Wire.beginTransmission(RTC_ADDRESS);
  Wire.write(0);
  Wire.endTransmission();

  Wire.requestFrom(RTC_ADDRESS, 7);
  uint8_t _second = bcd2bin(Wire.read() & 0x7F);
  uint8_t _minute = bcd2bin(Wire.read());
  uint8_t _hour = bcd2bin(Wire.read());
  Wire.read();
  uint8_t _day = bcd2bin(Wire.read());
  uint8_t _month = bcd2bin(Wire.read());
  uint16_t _year = bcd2bin(Wire.read()) + 2000;

  Serial.print(String(_year) + "/");
  Serial.print(String(_month) + "/");
  Serial.print(String(_day) + " ");

  Serial.print(String(_hour) + ":");
  Serial.print(String(_minute) + ":");
  Serial.println(String(_second));
}

void RTCinterrupt(){
  printTime = true;
}

void setup(){
  Serial.begin(9600);
  Wire.begin();
  attachInterrupt(0, RTCinterrupt, FALLING);
  timeSetting(2013, 4, 6, 15, 0, 0);
}

void loop(){
  if(printTime){
    printTime = false;

    printCurrentTime();
  }
}
```

■ 問題25–2

```
#include <LiquidCrystal.h>
#include <Wire.h>
#define RTC_ADDRESS 0x68

LiquidCrystal lcd(3, 4, 5, 6, 7, 8);
boolean printTime = false;

byte bin2bcd(int n){
  byte val = 0;

  int ten = n / 10;
  int one = n % 10;

  val = (ten << 4) | one;

  return val;
}

byte bcd2bin(byte n){
  byte val = 0;

  int ten = n >> 4;
  int one = n & 0x0F;

  val = ten * 10 + one;

  return val;
}

void timeSetting(int _year, int _month, int _day, int _hour, int _min, int _sec){
  Wire.beginTransmission(RTC_ADDRESS);
  Wire.write(0);
  Wire.write(bin2bcd(_sec));
  Wire.write(bin2bcd(_min));
  Wire.write(bin2bcd(_hour));
  Wire.write(0);
  Wire.write(bin2bcd(_day));
  Wire.write(bin2bcd(_month));
  Wire.write(bin2bcd(_year - 2000));

  Wire.write(0x10);
  Wire.endTransmission();
}

void printCurrentTime(){
  Wire.beginTransmission(RTC_ADDRESS);
  Wire.write(0);
  Wire.endTransmission();

  Wire.requestFrom(RTC_ADDRESS, 7);
  uint8_t _second = bcd2bin(Wire.read() & 0x7F);
  uint8_t _minute = bcd2bin(Wire.read());
  uint8_t _hour = bcd2bin(Wire.read());
  Wire.read();
  uint8_t _day = bcd2bin(Wire.read());
  uint8_t _month = bcd2bin(Wire.read());
  uint16_t _year = bcd2bin(Wire.read()) + 2000;

  char buffer1[11], buffer2[9];
  sprintf(buffer1, "%04d-%02d-%02d", _year, _month, _day);
  sprintf(buffer2, "%02d:%02d:%02d", _hour, _minute, _second);

  lcd.setCursor(0, 0);
  lcd.print(buffer1);
  lcd.setCursor(0, 1);
  lcd.print(buffer2);

  Serial.print(String(_year) + "/");
  Serial.print(String(_month) + "/");
  Serial.print(String(_day) + " ");

  Serial.print(String(_hour) + ":");
  Serial.print(String(_minute) + ":");
  Serial.println(String(_second));
}
```

```
void RTCinterrupt(){
  printTime = true;
}

void setup(){
  Serial.begin(9600);
  Wire.begin();
  lcd.begin(16, 2);
  attachInterrupt(0, RTCinterrupt, FALLING);
  timeSetting(2013, 4, 6, 15, 0, 0);
}

void loop(){
  if(printTime){
    printTime = false;

    printCurrentTime();
  }
}
```

問題 26-1

```
// Master
#include <Wire.h>

#define SLAVE 4
int button_pin = 7;

void setup() {
  Wire.begin();
  Serial.begin(9600);
}

void loop() {
  if(digitalRead(button_pin)){
    Wire.requestFrom(SLAVE, 1);
    Serial.println("Button Pressed...");
    delay(200);
  }
}

// Slave
#include <Wire.h>
#define SLAVE 4

int LED_pin = 13;
boolean state = false;

void setup() {
  Serial.begin(9600);
  Wire.begin(SLAVE);
  Wire.onRequest(sendToMaster);
  pinMode(LED_pin, OUTPUT);
}

void loop () {
}

void sendToMaster() {
  Serial.println("Data from the master arrived...");
  state = !state;
  digitalWrite(LED_pin, state);
}
```

問題 26-2

```
// Master
#include <SPI.h>

int button_pin = 7;

void setup (void)
```

```
{
  SPI.begin ();
  digitalWrite(SS, HIGH);

  SPI.setClockDivider(SPI_CLOCK_DIV16);
  Serial.begin(9600);
}

void loop() {
  if(digitalRead(button_pin)){
    digitalWrite(SS, LOW);
    SPI.transfer('1');
    digitalWrite(SS, HIGH);

    Serial.println("Button Pressed...");
    delay(200);
  }
}

// Slave
#include <SPI.h>

int LED_pin = 2;
boolean state = false;
boolean process_it = false;

void setup (void)
{
  Serial.begin (9600);

  pinMode(MISO, OUTPUT);
  SPI.setClockDivider(SPI_CLOCK_DIV16);
  SPCR |= _BV(SPE);
  SPCR &= ~_BV(MSTR);
  SPCR |= _BV(SPIE);

  pinMode(LED_pin, OUTPUT);
}

ISR (SPI_STC_vect)
{
  byte c = SPDR;
  process_it = true;
}

void loop () {
  if(process_it){
    process_it = false;
    Serial.println("Data from the master arrived...");
    state = !state;
    digitalWrite(LED_pin, state);
  }
}
```

■ 問題 27-2

```
int cols[] = {2, 3, 4};
int rows[] = {5, 6, 7};

void setup(){
  for(int i = 0; i < 3; i++){
    pinMode(cols[i], OUTPUT);
    pinMode(rows[i], INPUT);
  }

  Serial.begin(9600);
}

void loop(){
  int input[3][3] = {0}, i, j, c;

  for(c = 0; c < 3; c++){
    for(i = 0; i < 3; i++){
      if(c == digitalWrite(cols[i], HIGH);
      else digitalWrite(cols[i], LOW);
```

```
    }

    for(i = 0; i < 3; i++)
      input[i][c] = digitalRead(rows[i]);
  }

  Serial.println("=====");
  for(i = 0; i < 3; i++){
    for(j = 0; j < 3; j++){
      if(input[i][j] == HIGH) Serial.print("O");
      else Serial.print(".");
    }
    Serial.println();
  }

  delay(1000);
}
```

■ 問題 27-3

```
#include <Wire.h>
#include <Adafruit_MCP23017.h>

Adafruit_MCP23017 mcp;

void setup() {
  mcp.begin();

  for(int i = 0; i < 8; i++){
    mcp.pinMode(i, INPUT);
    mcp.pullUp(i, HIGH);
  }
  for(int i = 8; i < 16; i++){
    mcp.pinMode(i, OUTPUT);
  }

  Serial.begin(9600);
}

void loop() {
  for(int i = 0; i < 8; i++){
    boolean buttonState = mcp.digitalRead(i);

    if(buttonState) mcp.digitalWrite(i + 8, LOW);
    else mcp.digitalWrite(i + 8, HIGH);
  }

  delay(200);
}
```

■ 問題 28-1

```
#include <EEPROM.h>

typedef union{
  float f;
  byte b[4];
}EEPROM_float;

float ReadFloatFromEEPROM(int address){
  byte value;
  EEPROM_float f;

  f.b[0] = EEPROM.read(address);
  f.b[1] = EEPROM.read(address + 1);
  f.b[2] = EEPROM.read(address + 2);
  f.b[3] = EEPROM.read(address + 3);

  return f.f;
}

void WriteFloatToEEPROM(int address, float value){
  EEPROM_float f;
```

```
  f.f = value;

  EEPROM.write(address, f.b[0]);
  EEPROM.write(address + 1, f.b[1]);
  EEPROM.write(address + 2, f.b[2]);
  EEPROM.write(address + 3, f.b[3]);
}

void setup(){
  Serial.begin(9600);
  WriteFloatToEEPROM(10, 3.14);
  float f = ReadFloatFromEEPROM(10);
  Serial.println(f);
}

void loop(){
}
```

■ 問題 29-1

```
#include <Metro.h>

Metro timer_1sec(500);
Metro timer_2sec(1000);

int LED1 = 10, LED2 = 11;
boolean state1 = false, state2 = false;

void setup(){
  Serial.begin(9600);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
}

void loop(){
  if(timer_1sec.check()){
    Serial.println(">> Timer 1 has expired.");
    state1 = !state1;
    digitalWrite(LED1, state1);
  }
  if(timer_2sec.check()){
    Serial.println("** Timer 2 has expired.");
    state2 = !state2;
    digitalWrite(LED2, state2);
  }
}
```

■ 問題 29-2

```
#include <MsTimer2.h>

int index = 0;
boolean process = false;
int pins[] = {10, 11, 12, 13};

void LEDFunction(){
  process = true;
  index = (index + 1) % 4;
}

void setup(){
  Serial.begin(9600);
  for(int i = 0; i < 4; i++)
    pinMode(pins[i], OUTPUT);

  MsTimer2::set(500, LEDFunction);
  MsTimer2::start();
}

void loop(){
  if(process){
    process = false;
    for(int i = 0; i < 4; i++){
```

```
      if(i == index) digitalWrite(pins[i], HIGH);
      else digitalWrite(pins[i], LOW);
    }
  }
}
```

■ **問題 30-1**

```
#include <SPI.h>
#include <SD.h>

#define SD_CS 4

File start_dir;

void setup()
{
  Serial.begin(9600);
  SD.begin(SD_CS);

  char fileName[] = "check.txt";

  if(SD.exists(fileName)){
    Serial.println(String(fileName) + " exists...");
    Serial.print(String(" try to delete ") + fileName + " file...");
    boolean result = SD.remove(fileName);
    Serial.println(String("-> ") + (result ? "deleted" : "deletion failed"));
  }
  else{
    Serial.println(String(fileName) + " does not exist...");
    Serial.print(String(" try to create ") + fileName + " file...");
    File newFile = SD.open(fileName, FILE_WRITE);
    Serial.println(String("-> ") + (newFile ? "created" : "creation failed"));
    newFile.close();
  }
}

void loop()
{
}
```

■ **問題 30-2**

```
#include <SPI.h>
#include <SD.h>
#include <core_build_options.h>
#include <swRTC.h>

#define SD_CS 4

File file;
swRTC rtc;
int button_pin = 3;

void setup()
{
  Serial.begin(9600);

  rtc.stopRTC();
  rtc.setTime(15, 0, 0);
  rtc.setDate(4, 6, 2013);
  rtc.startRTC();

  SD.begin(SD_CS);
  pinMode(button_pin, INPUT);
}

void loop()
{
  if(digitalRead(button_pin)){
    char buffer[21];
    sprintf(buffer, "%04d-%02d-%02d %02d:%02d:%02d\n",
      rtc.getYear(), rtc.getMonth(), rtc.getDay(),
      rtc.getHours(), rtc.getMinutes(), rtc.getSeconds());
    Serial.print(buffer);
```

```
    file = SD.open("button.txt", FILE_WRITE);
    file.println(buffer);
    file.close();

    delay(200);
  }
}
```

■ 問題 33-1

```
int incomingByte = 0;
boolean state1 = false, state2 = false;
int pin1 = 24, pin2 = 25;

void setup() {
  Serial.begin(9600);
  pinMode(pin1, OUTPUT);
  pinMode(pin2, OUTPUT);
}

void loop() {
  if (Serial.available() > 0) {
    incomingByte = Serial.read();

    if(incomingByte == '1'){
      Serial.println("Pressed 1");
      state1 = ~state1;
      digitalWrite(pin1, state1);
    }
    if(incomingByte == '2'){
      Serial.println("Pressed 2");
      state2 = ~state2;
      digitalWrite(pin2, state2);
    }
  }
}
```

■ 問題 33-2

```
int pins[] = {8, 9, 10, 11, 12, 13, 14, 15};  // PORT B

void setup(){
  for(int i = 0; i < 8; i++){
    pinMode(pins[i], OUTPUT);
  }
}

void loop(){
  int analog = analogRead(A0);                // PORT Fの PIN 0 (PF0)に可変抵抗の接続
  int LED_no = map(analog, 0, 1023, 0, 8);    // 0から8の間の値に変換

  for(int i = 0; i < LED_no; i++)
    digitalWrite(pins[i], HIGH);

  for(int i = LED_no; i < 8; i++)
    digitalWrite(pins[i], LOW);
}
```

■ 問題 34-1

```
int LED_pin = 3;
int switch_pin = 4;

void setup(){
  pinMode(LED_pin, OUTPUT);
  pinMode(switch_pin, INPUT);
}

void loop(){
  digitalWrite(LED_pin, digitalRead(switch_pin));

  delay(100);
}
```

■ **問題 34-2**

```
int LED_pin = 3;
int switch_pin = 4;

int state = LOW;

void setup(){
  pinMode(LED_pin, OUTPUT);
  pinMode(switch_pin, INPUT);
}

void loop(){
  if(digitalRead(switch_pin) == HIGH){
    if(state == HIGH) state = LOW;
    else state = HIGH;

    digitalWrite(LED_pin, state);

    delay(200);
  }
}
```