

# ATMEGA MICROCONTROLLER MINIMUM SYSTEM

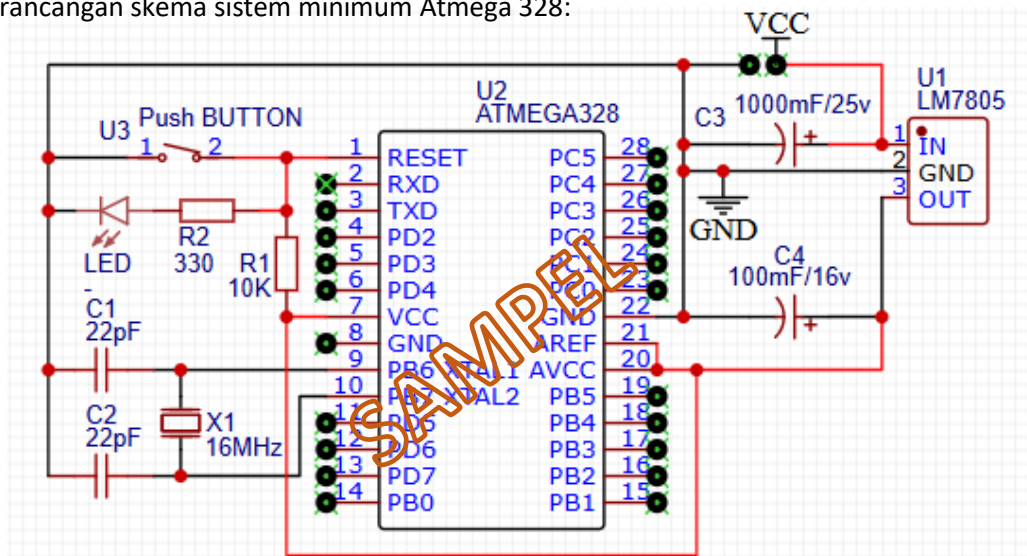
Program Studi: SISTEM KOMPUTER (IIB DARMAJAYA)

Matakuliah : Desain dan Simulasi Rangkaian Elektronika (5SK1)

Sistem minimum dibangun untuk menyediakan rangkaian dasar minimal mikrokontroler agar berfungsi menjalankan program sehingga dapat diaplikasikan pada proyek elektronika seperti sistem sensor, kontrol otomatis, dan otomatisasi lainnya tanpa perlu **Modul Arduino lengkap**. Langkah pembuatan:

## 1. MINIMUM SYSTEM SCHEMATIC DESIGN

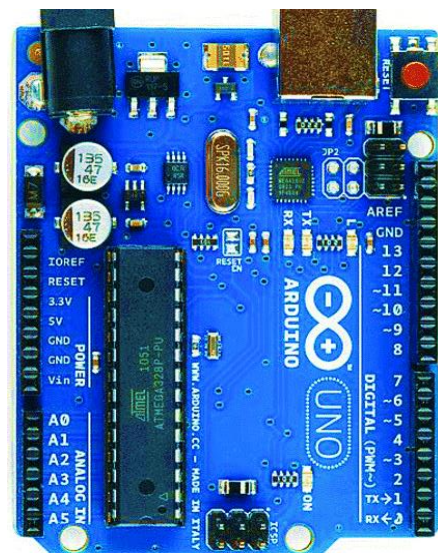
Perancangan skema sistem minimum Atmega 328:



Gambar 1. Contoh Skema minimum Sistem Atmega 328

ATmega328P pin mapping			
Arduino function	Arduino function		
reset	PC6 1	PC5 28	analog input 5
digital pin 0 <b>RX</b>	PD0 2	PC4 27	analog input 4
digital pin 1 <b>TX</b>	PD1 3	PC3 26	analog input 3
digital pin 2	PD2 4	PC2 25	analog input 2
digital pin 3 <b>PWM</b>	PD3 5	PC1 24	analog input 1
digital pin 4	PD4 6	PC0 23	analog input 0
VCC	VCC 7	GND 22	GND
GND	GND 8	AREF 21	analog reference
crystal	PB6 9	AVCC 28	AVCC
crystal	PB7 10	PB5 19	<b>SCK</b> digital pin 13
digital pin 5 <b>PWM</b>	PD5 11	PB4 18	<b>MISO</b> digital pin 12
digital pin 6 <b>PWM</b>	PD6 12	PB3 17	<b>PWM</b> digital pin 11
digital pin 7	PD7 13	PB2 16	<b>PWM</b> digital pin 10
digital pin 8	PB0 14	PB1 15	<b>PWM</b> digital pin 9

When using ISP to the chip



Gambar 2. Mapping Pin OUT IC Atmega 328 dan Modul Arduino UNO

Peralatan yang digunakan:

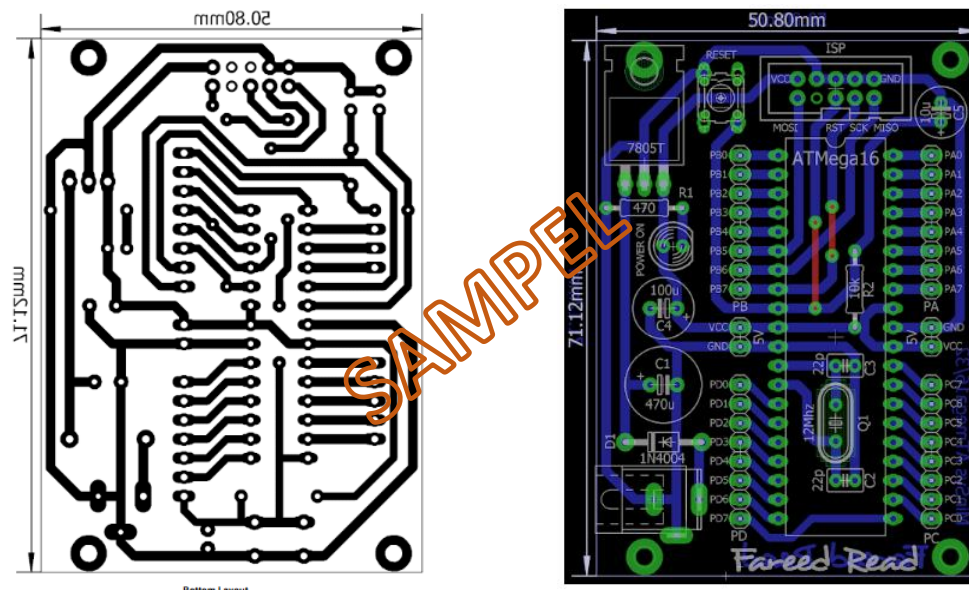
- Tools Set (tang, obeng, dll)
- Bor Tangan (drill 0,5mm)
- Solder, timah, Nampan Plastik
- Sarung Tangan waterproof
- Lotion anti nyamuk, Amplas halus

Bahan dan Komponen yang dibutuhkan:

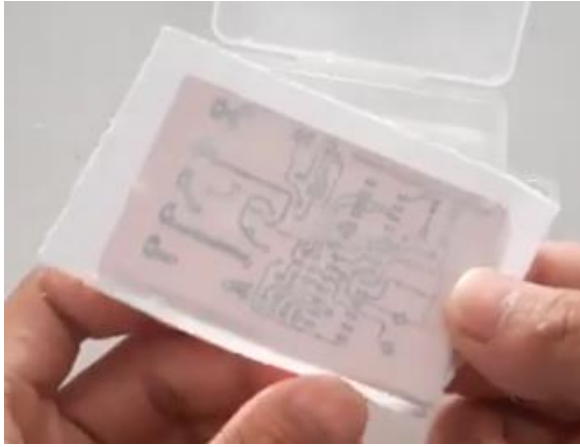
a) Papan PCB	Polos	(8x6cm)
b) Larutan feriklorit	feri klorida, $FeCl_3$	1 bks (200gr)
c) IC + Socket 28 pin	Atmega328	1
d) Condensator	Crystal 16Mhz	1
e) Capacitor	22Pf	2
f) Push Button	Momentary (takticle)	4
g) Resistor	330 $\Omega$	4
h) Resistor	10 K $\Omega$	1
i) Lampu LED	LED 3mm atau 5mm	1
j) Konektor Power DC	micro USB / DC 5mm	1
k) Pin Header	male atau female	1x20
l) IC Regulator	7805	1
m) Elco	1000/25v & 100/16v	1

## 2. PCB ASSEMBLY

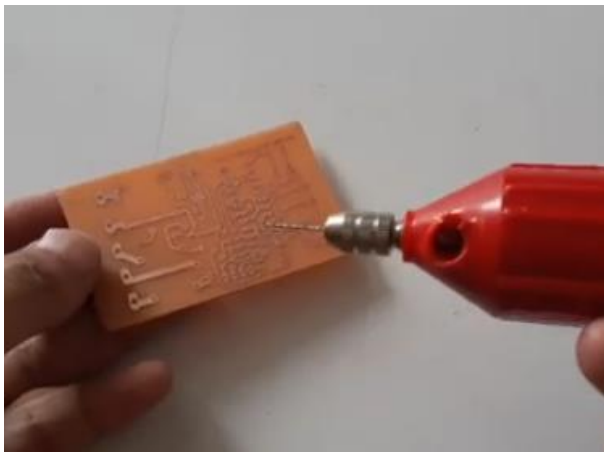
Pembuatan PCB bertujuan untuk mentransformasi desain skema rangkaian yang abstrak menjadi bentuk fisik yang fungsional dalam menghubungkan komponen elektronik secara rapi, efisien, dan andal, menggantikan metode perakitan kabel manual yang rumit dan berantakan.



Gambar 3. Contoh Skema PCB Tampak Bawah dan Tampak Atas



Gambar 4. Proses Penempelan Skema Ke papan PCB dan Perlarutan PCB  
PCB Assembly → ( <https://www.youtube.com/watch?v=sCqOYDUKtYg> )



Gambar 5. Proses Pengeboran dan Pemasangan Komponen (Soldering)

### 3. COMPONENT ASSEMBLY (TEST DOWNLOADER)

Setelah semua komponen terpasang, langkah selanjutnya inject firmware (burning downloader) ke Minimum Sistem. **Peralatan yang Dibutuhkan:**

- **Sistem Minimum ATmega328:** yang telah dibuat.
- **Programmer ISP:** Bisa berupa Arduino Uno/Nano/Mega lain atau programmer khusus seperti USBasp/USBtinyISP.
- **Kabel Jumper:** Untuk menghubungkan programmer ISP ke sistem minimum yang telah dibuat.
- **Komputer:** Terinstal software Arduino IDE.

#### a) Tahapan Burning Bootloader:

##### Menggunakan Arduino sebagai Programmer ISP (Metode Umum):

- Hubungkan Arduino yang akan digunakan sebagai programmer ISP (misalnya Arduino Uno) ke komputer melalui kabel USB.
- Buka **Arduino IDE**, pilih board dan port yang sesuai.

- Buka contoh program: **File > Examples > ArduinoISP** dan *Upload* program "ArduinoISP" ini ke board programmer.

**b) Hubungkan Programmer ke Sistem Minimum**

Hubungkan pin-pin pada Arduino programmer ke sistem minimum ATmega328 Anda menggunakan kabel jumper, berdasarkan koneksi ISP (SPI):

1. **VCC** (5V) ke pin VCC ATmega328P.
2. **GND** ke pin GND ATmega328P.
3. **Pin 10** (pada programmer) ke pin **Reset** (pin 1) ATmega328.
4. **Pin 11** (MOSI pada programmer) ke pin **MOSI** (pin 17) ATmega328.
5. **Pin 12** (MISO pada programmer) ke pin **MISO** (pin 18) ATmega328.
6. **Pin 13** (SCK pada programmer) ke pin **SCK** (pin 19) ATmega328.

**c) Lakukan Proses Burning:**

1. Pada Arduino IDE, pastikan Anda memilih **Board** yang benar untuk **sistem minimum target** (misalnya, "Arduino Uno" atau "Arduino Nano").
2. Pilih board **Programmer** menjadi "**Arduino as ISP**" (atau "AVRISP mkII" tergantung versi IDE).
3. Pilih port Serial yang benar untuk **board programmer** (bukan board target).
4. Klik menu **Tools > Burn Bootloader**. Jika berhasil, di bilah status Arduino IDE akan muncul pesan "**Done burning bootloader**".

```

ArduinoISP.ino
---
104 #endif
105
106 // By default, use hardware SPI pins:
107 #ifndef ARDUINOISP_PIN_MOSI
108 #define ARDUINOISP_PIN_MOSI MOSI
109 #endif
110
111 #ifndef ARDUINOISP_PIN_MISO
112 #define ARDUINOISP_PIN_MISO MISO
113 #endif
114
115 #ifndef ARDUINOISP_PIN_SCK
116 #define ARDUINOISP_PIN_SCK SCK
117 #endif
118
119 // Force bitbanged SPI if not using the hardware SPI pins:
120 #if (ARDUINOISP_PIN_MISO != MISO) || (ARDUINOISP_PIN_MOSI != MOSI) || (ARDUINOISP_PIN_SCK != SCK)
121 #undef USE_HARDWARE_SPI
122 #endif
123
124
125 // Configure the serial port to use.
126 //

```

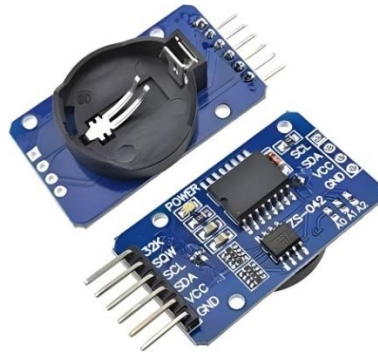
Gambar 6. Upload Program Burning Downloader (Arduino ISP)  
 Cara burn bootloader atmega328 → ( <https://www.youtube.com/watch?v=Rn6vzulFqVg> )

#### 4. SYSTEM DEVELOPMENT WITH MINSIS

Buat satu sistem (jam digital) menggunakan 7 segment atau modul LCD16x2. Untuk memastikan sistem minimum (sismin) ATmega328P dapat **berfungsi dan beroperasi secara mandiri** tanpa board Arduino dengan tambahan 3 tombol push button untuk setting jam.

Peralatan/Modul yang digunakan:

- a. Modul Utama: RTC DS3231



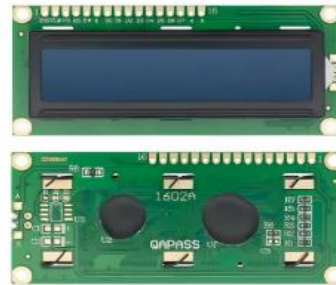
Gambar 7. Modul RTC DS3231

- b. Modul Pilihan: 7 Segment TM 1637



Gambar 8. Modul TM 1637

- c. Atau Modul LCD 16x2



Gambar 9. Modul LCD Display 16x2



Gambar 10. Contoh gambar rakitan Minimum Sistem ATMEGA328 (Jam Digital)

## CONTOH KODE PROGRAM

**Minimum Sistem, RTC DS3231 dan 7 Segment TM1637**  
**Dengan tambahan 3 tombol push button untuk setting pengaturan waktu (jam)**

---

```
#include <RTCLib.h>
#include <TM1637Display.h>
RTC_DS3231 rtc;
TM1637Display display(3, 2); // CLK, DIO

// Tombol
#define BTN_PLUS 4
#define BTN_SET 5
#define BTN_MINUS 6

// Variabel SET
bool isSetting = false;
uint8_t setMode = 0; // 0=jam, 1=menit
int setHour, setMinute;

// Blink untuk digit yang diset
bool blinkState = true;
unsigned long lastBlink = 0;

// Detik untuk kolom titik tengah
int curSecond = 0;

void setup() {
  Serial.begin(9600);
  pinMode(BTN_PLUS, INPUT_PULLUP);
  pinMode(BTN_SET, INPUT_PULLUP);
  pinMode(BTN_MINUS, INPUT_PULLUP);
  display.setBrightness(7);

  if (!rtc.begin()) {
    Serial.println("RTC not found!");
    while (1);
  }

  if (rtc.lostPower()) {
    //rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    rtc.adjust(DateTime(2026, 01, 1, 00, 00, 0));
    // tahun, bulan, tanggal, jam, menit, detik
  }
}

void loop() {

  // ----- MODE SET -----
  if (isSetting) {
    handleSettingMode();
    return;
  }
}
```

```

// ----- MODE NORMAL -----
DateTime now = rtc.now();
int hour = now.hour();
int minute = now.minute();
curSecond = now.second();

// Colon blink berdasarkan detik
uint8_t colonMask = (curSecond % 2 == 0) ? 0b01000000 : 0;
int toDisplay = hour * 100 + minute;
display.showNumberDecEx(toDisplay, colonMask, true);
// Masuk SET
if (buttonPressed(BTN_SET)) {
    isSetting = true;
    setMode = 0;
    setHour = hour;
    setMinute = minute;
    delay(200);
}
}

void handleSettingMode() {
    // Blink timer
    if (millis() - lastBlink > 500) {
        lastBlink = millis();
        blinkState = !blinkState;
    }
    // Tombol +
    if (buttonPressed(BTN_PLUS)) {
        if (setMode == 0) setHour = (setHour + 1) % 24;
        else setMinute = (setMinute + 1) % 60;
    }
    // Tombol -
    if (buttonPressed(BTN_MINUS)) {
        if (setMode == 0) setHour = (setHour == 0 ? 23 : setHour - 1);
        else setMinute = (setMinute == 0 ? 59 : setMinute - 1);
    }
    // Tombol SET → pindah tahap
    if (buttonPressed(BTN_SET)) {
        setMode++;
        if (setMode > 1) {
            rtc.adjust(DateTime(2025, 1, 1, setHour, setMinute, 0));
            isSetting = false;
            return;
        }
    }
}

uint8_t segs[4];

// =====
//   SETTING JAM
if (setMode == 0) {

    if (blinkState) {
        // Tampilkan nilai normal (colon OFF)
        display.showNumberDecEx(setHour * 100 + setMinute, 0, true);
    }
}

```

```

    return;
} else {
    // Blink jam → --MM
    segs[0] = 0x40; // '-'
    segs[1] = 0x40; // '-'
    segs[2] = display.encodeDigit(setMinute / 10);
    segs[3] = display.encodeDigit(setMinute % 10);
    display.setSegments(segs);
    return;
}
}

// =====
//   SETTING MENIT

if (setMode == 1) {

    if (blinkState) {
        display.showNumberDecEx(setHour * 100 + setMinute, 0, true);
        return;
    } else {
        // Blink menit → HH--
        segs[0] = display.encodeDigit(setHour / 10);
        segs[1] = display.encodeDigit(setHour % 10);
        segs[2] = 0x40; // '-'
        segs[3] = 0x40; // '-'
        display.setSegments(segs);
        return;
    }
}

bool buttonPressed(int pin) {
    if (digitalRead(pin) == LOW) {
        delay(20);
        if (digitalRead(pin) == LOW) return true;
    }
    return false;
}

```

**DEVELOPMENT REPORT CARD**  
**ATMEGA MICROCONTROLLER MINIMUM SYSTEM**

Program Studi: SISTEM KOMPUTER (IIB DARMAJAYA)

NPM :

NAME :

Matakuliah : **Desain dan Simulasi Rangkaian Elektronika**

Subject : *(Electronic Circuit Design and Simulation)*

Week	Phases / Progress	Achievement (EV%)	Date	Sign
1	<b>Minimum System Schematic Design</b>			
	* Schematic Design ATMEGA 8/16/32/328 (MinSis)			
	Path Design & Component Layout			
	Print Out Design (Results)			
2, 3	<b>PCB Assembly</b>			
	Etching (Pelarutan PCB)			
	Drilling (Pengeboran PCB)			
	Circuit Checking (Pengecekan Jalur PCB)			
4, 5	<b>Component Assembly</b>			
	Soldering (Penyolderan Komponen)			
	Component Checking (Pengecekan Pemasangan)			
	Testing Sistem Downloader (Blinking LED)			
6, 7	<b>System Development with MinSis</b>			
	RTC Modul (DS3231)			
	* 7 Segment (TM1637) / LCD Display Modul 16x2			
	* In addition (modul):			
8	<b>Simulation and Presentation</b>			

Notes : \* Cross out one option