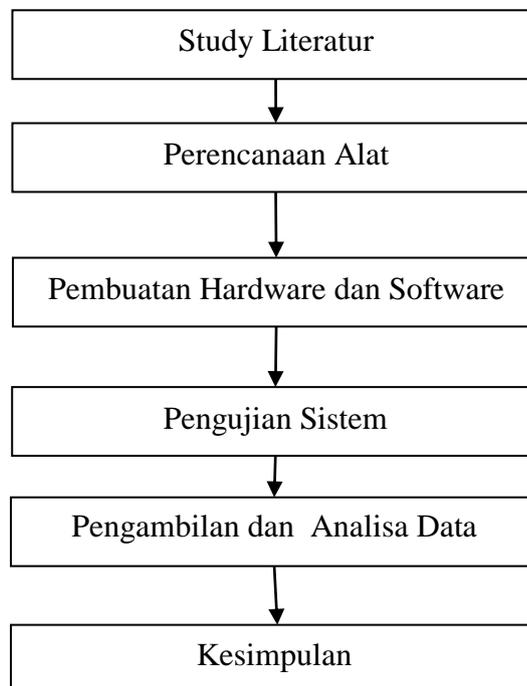


BAB III

METODOLOGI PENELITIAN

Dalam bab ini akan di bahas alur proses penyelesaian masalah, metode dan prosedur yang digunakan untuk menyelesaikan permasalahan. Dapat dilihat seperti pada gambar 3.1. :



Gambar 3.1 Metodologi Penelitian

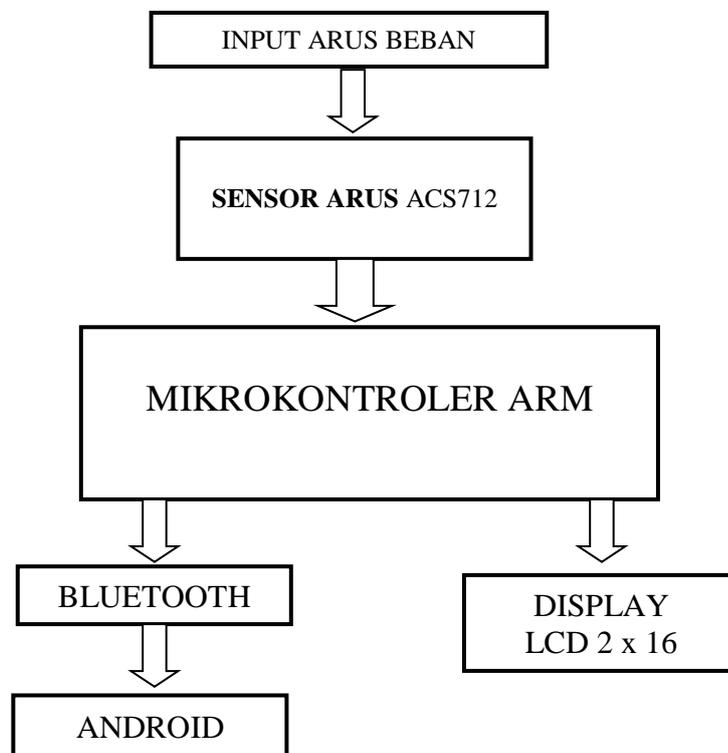
3.1. Study Literatur

Dalam perancangan pembuatan alat ini dibutuhkan sumber – sumber refrensi sebagai bahan acuan dan pertimbangan, sumber refrensi didapatkan dari sumber langsung dan tak langsung, Sumber langsung di dapatkan dari diskusi atau konsultasi dengan dosen, sedangkan sumber tak langsung didapat dari tulisan

laporan penelitian–penelitian yang dilakukan sebelumnya, buku, internet serta referensi–referensi lain yang berkaitan perancangan dan pembuatan alat.

3.2. Perencanaan Alat

Pada Proposal Skripsi ini akan di buat alat monitoring arus beban yang tersalurkan pada Gardu Induk dengan menggunakan Bluetooth HC-05 berbasis mikrokontroler ARM.



Gambar 3.2 Sistem monitoring arus beban pada gardu induk dengan Android

menggunakan Bluetooth berbasis mikrokontroler.

Fungsi dari block diagram di atas adalah sebagai berikut:

1. Input arus beban

Merupakan jumlah muatan listrik yang mengalir tiap satuan waktu.

2. Sensor Arus

Berfungsi sebagai pendeteksi besaran arus yang disalurkan.

3. Mikrokontroller ARM

Berfungsi sebagai pengolah data yang di deteksi oleh sensor arus dan memproses data yang diterima untuk diteruskan melalui bluetooth.

4. Display

Berfungsi untuk menampilkan Besaran arus yang tersalurkan pada peralatan.

5. Bluetooth.

Berfungsi sebagai sarana komunikasi untuk mengirimkan data dari mikrokontroler ke perangkat android.

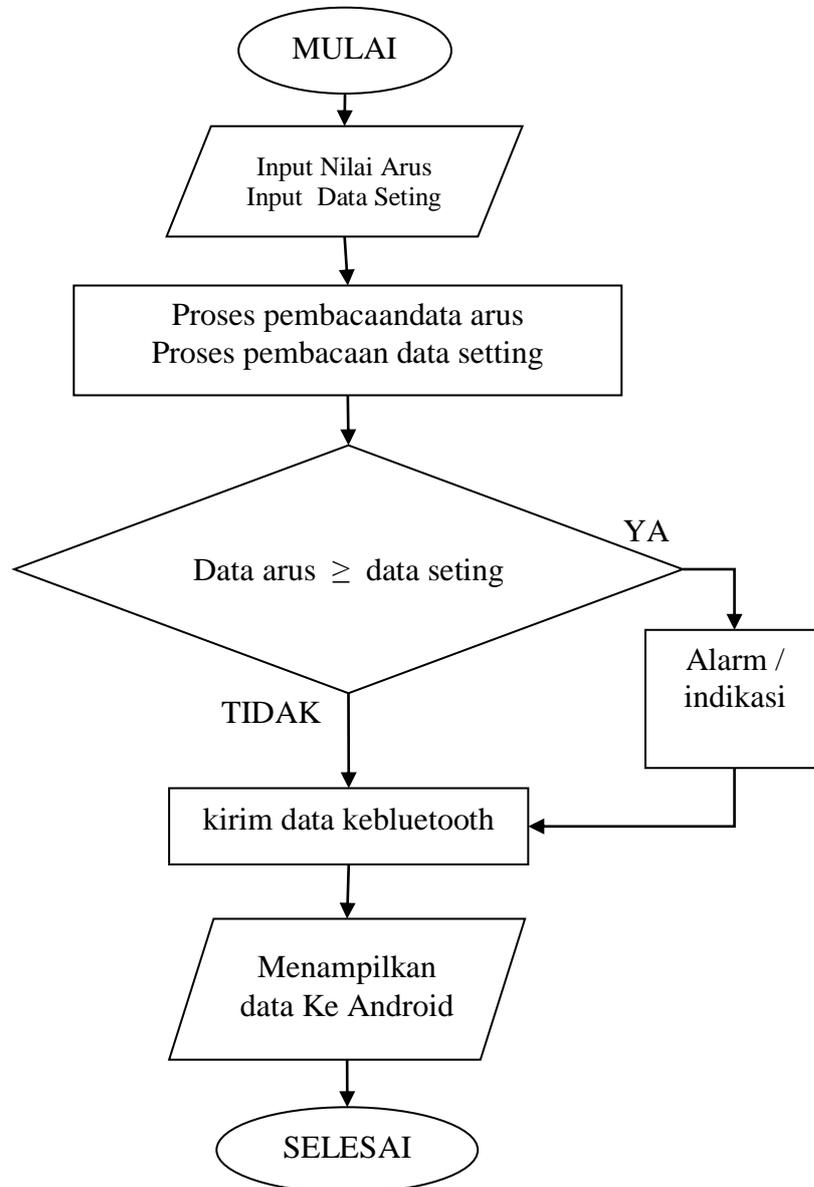
6. Android

Berfungsi sebagai divice penerima data dari modul bluetooth yang ada pada mikrokontroler

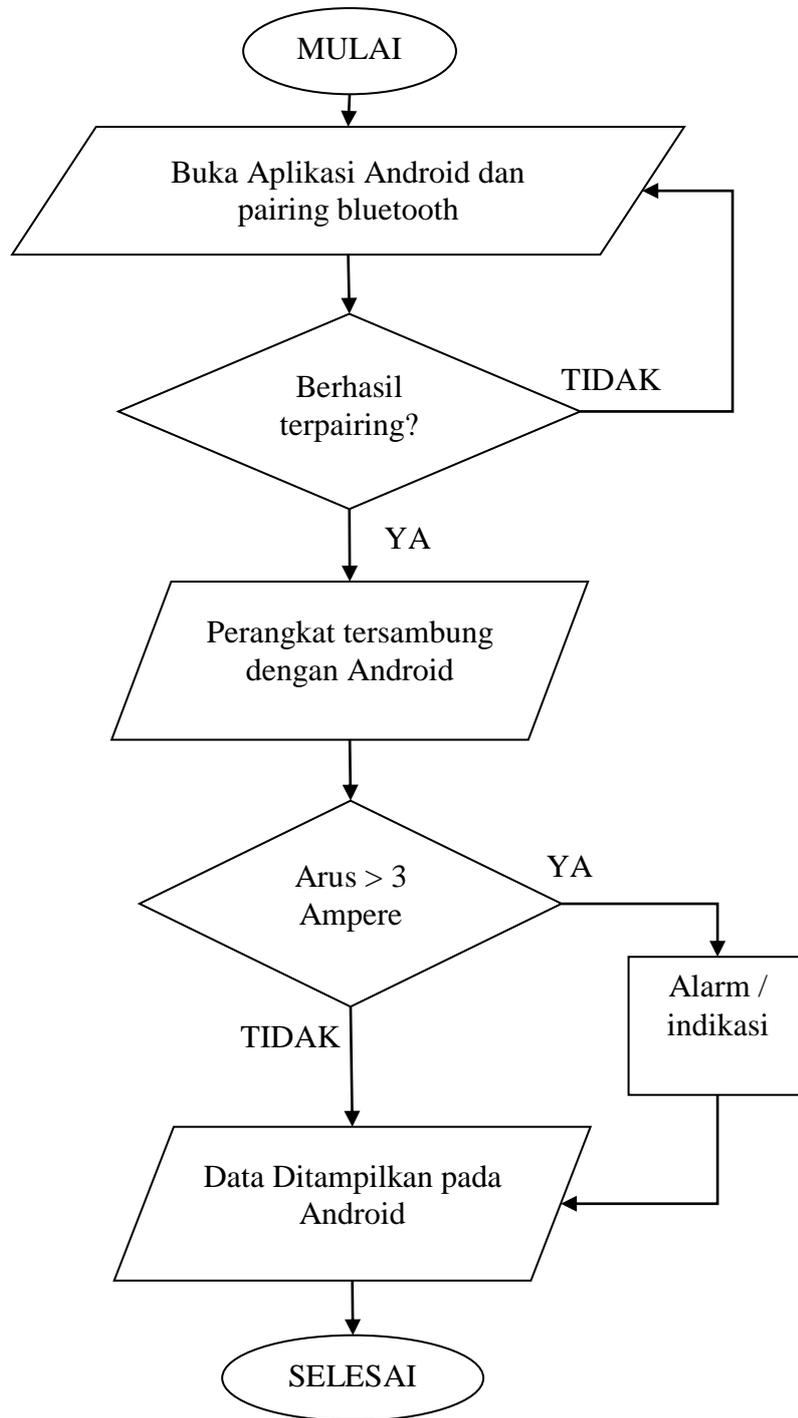
3.3 Perancangan Software

Pada perancangan *software* monitoring arus beban yang tersalurkan pada gardu induk ini menggunakan bahasa C dan program *Cocox_CoIDE* sebagai *compilernya*. Bahasa C digunakan untuk membuat program ini dibutuhkan untuk mengatur kinerja dari alat sehingga dapat berjalan sesuai dengan yang diharapkan. *Cocox_CoIDE* digunakan sebagai alat bantu pemrograman (*programming toll*) yang bekerja dalam lingkungan pengembangan perangkat lunak (*software*) yang terintegrasi (*Intregrated Development Enviroment, IDE*). Sedangkan software

yang digunakan untuk membuat aplikasi android monitoring adalah program *APP Inventor*.



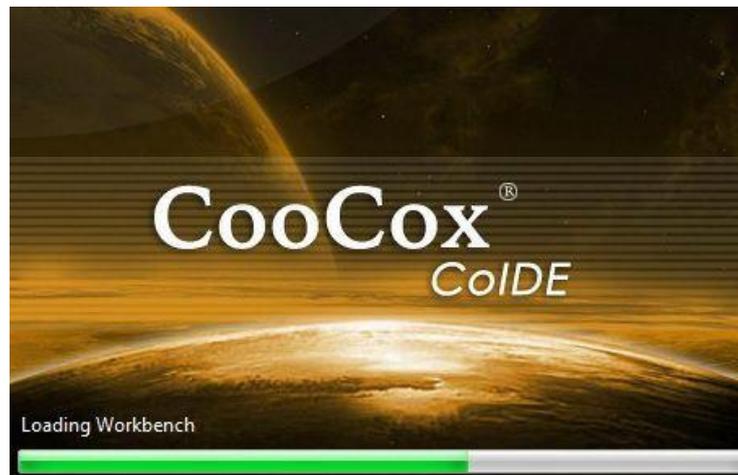
Gambar 3.3 Flowchart Sistem Monitoring



Gambar 3.4 Flowchart Monitoring Android

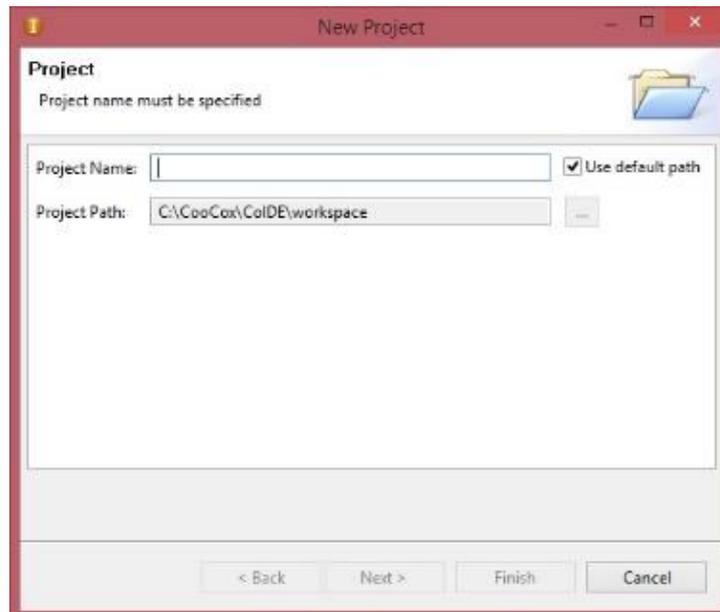
3.3.1 Pemrograman Menggunakan Coocox_CoIDE

Coocox_CoIDE merupakan sistem *software C-cross compiler*, dimana program dapat ditulis menggunakan bahasa C. Dengan menggunakan pemograman bahasa bahasa C diharapkan waktu disain (*deleloping time*) akan menjadi lebih singkat. Setelah program dalam bahasa C ditulis dan dilakukan kompilasi tidak terdapat kesalahan/*error*, maka proses download dapat dilakukan. Mikrokontroler ARM mendukung sistem *download* secara ISP (*In-System Programing*).



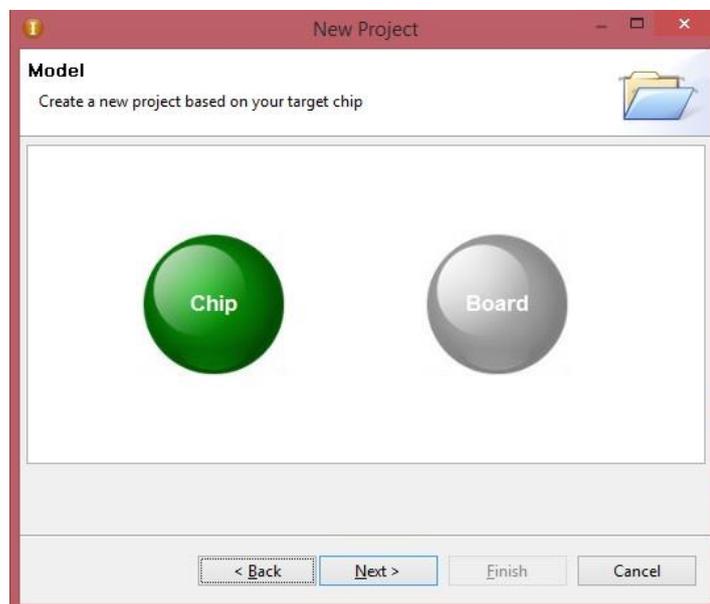
Gambar 3.5. Tampilan *Coocox_CoIDE*

Untuk memulai pemograman menggunakan CodeVisionAVR pilih pada menu **Project>New Project** kemudian akan muncul kotak dialog pada gambar 3.6 di bawh ini :



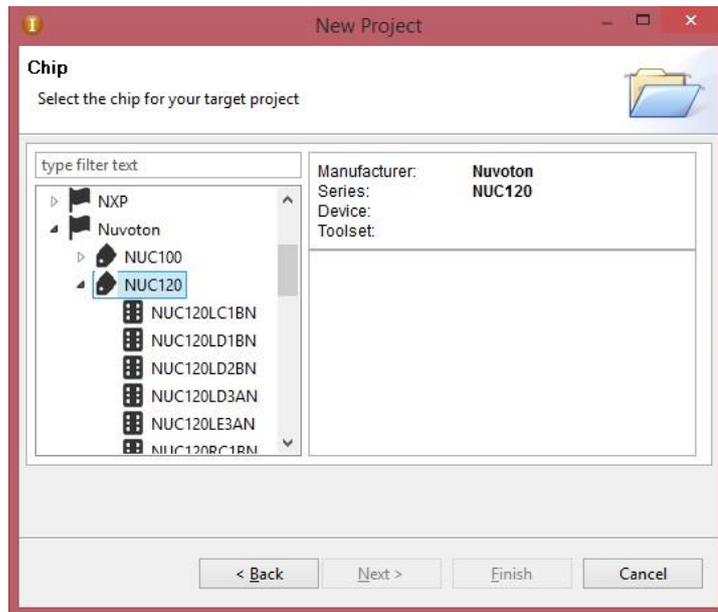
Gambar 3.6. Mengisi Project Name

Isilah *Project Name* sesuai dengan keinginan kemudian tekan *Next>*, maka akan muncul kotak dialog pada gambar 3.7 berikut



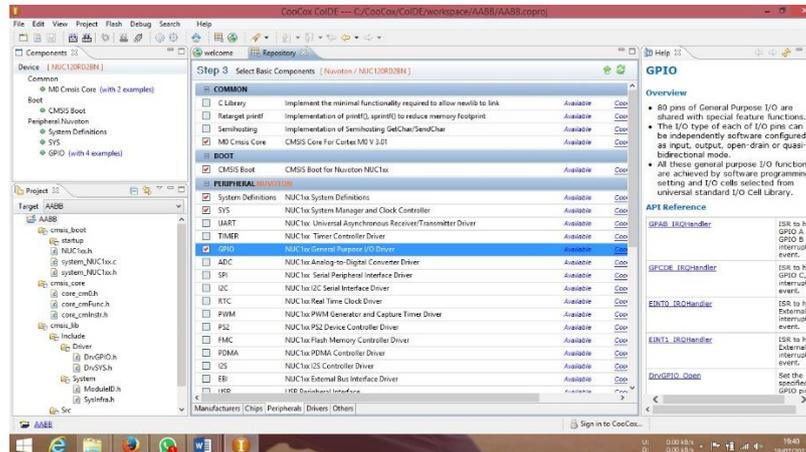
Gambar 3.7. Tampilan Model

Pilihlah Model *Chip* kemudian tekan *Next>*, maka akan muncul kotak dialog pada gambar 3.8 berikut



Gambar 3.8. Tampilan Model Chip

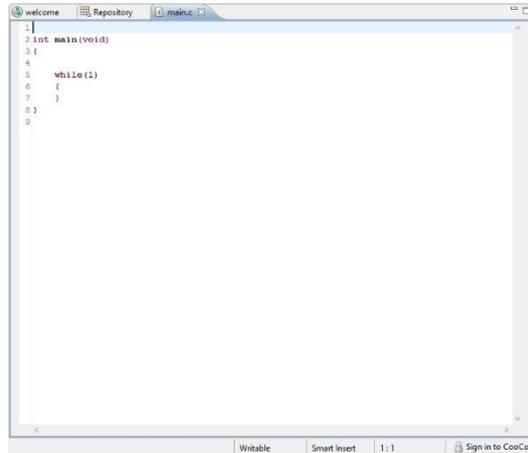
Pilihlah Model *Chip* **Nuvoton>NUC120>NUC120VRD2BN** sesuai dengan model chip yang digunakan kemudian tekan **Finish**, maka akan muncul jendela repository seperti pada gambar 3.9 berikut.



Gambar 3.9. Tampilan jendela Repository

Pilihlah **Peripheral.Nuvoton** dan Centang pada **GPIO**, maka dengan otomatis menu yang lainya juga akan ikut ter cntang seperti gambar di atas. Selanjutnya pada tab **Project** geser kursor sampai ke paling bawah dan klik 2x

pada tulisan **main.c** maka akan muncul kotak jendela **main.c** seperti pada gambar 3.10 berikut



Gambar 3.10. Tampilan jendela main.c

Pada tab jendela main.c inilah tuliskan program bahasa C dan setting yang akan kita gunakan untuk monitoring arus beban pada Gardu induk dengan Android menggunakan bluetooth.

List programnya sebagai berikut :

```
#include "DrvGPIO.h"
#include "DrvUART.h"
#include "DrvSYS.h"
#include "NUC1xx.h"
#include "DrvADC.h"
#include <stdio.h>

void UART0_INT_HANDLE(uint32_t u32IntStatus);
unsigned char TEXT;
unsigned char buf[1]={0xFF};

STR_UART_T      UartParam;
E_DRVGPIO_FUNC  FuncNum = E_FUNC_UART0;
E_UART_PORT     UartNum = UART_PORT0;

void initUART()
{
    UNLOCKREG();
    SYSCLK->PWRCON.XTL12M_EN = 1;
    DrvSYS_SetOscCtrl(E_SYS_XTL12M, ENABLE);
    DrvSYS_SelectIPClockSource(E_SYS_UART_CLKSRC, 0);

    SysTick->LOAD = 50000 * 22;
```

```

SysTick->VAL = (0x00);
SysTick->CTRL = (1 << 2) | (1<<0);

while((SysTick->CTRL & (1 << 16)) == 0);
SYS->GPBMFP.UART0_RX =1;
SYS->GPBMFP.UART0_TX =1;
SYS->GPBMFP.UART0_nRTS_nWRL =1;
SYS->GPBMFP.UART0_nCTS_nWRH =1;
UartParam.u32BaudRate = 9600;
UartParam.u8cDataBits = DRVUART_DATABITS_8;
UartParam.u8cStopBits = DRVUART_STOPBITS_1;
UartParam.u8cParity = DRVUART_PARITY_NONE;
UartParam.u8cRxTriggerLevel = DRVUART_FIFO_1BYTES;
UartParam.u8TimeOut = 0;

DrvGPIO_InitFunction(FuncNum);
DrvUART_Open(UartNum, &UartParam);
DrvUART_EnableInt(UartNum, (DRVUART_RLSINT |
DRVUART_RDAINT), UART0_INT_HANDLE);
}

void UART0_INT_HANDLE(uint32_t u32IntStatus) {
    if(u32IntStatus & DRVUART_RDAINT) {

        while (UART0->ISR.RDA_IF==1)

        {
            DrvUART_Read(UartNum,buf,1);

        }
        TEXT=buf[0];

    }
}

int main(void) {
    uint8_t i = 0;

    char TEXT1[16];

    uint16_t tampung1;

    uint16_t arus1;

    uint16_t bacal;

    uint8_t tampil1[5];

    uint8_t mode = 0;

    DrvGPIO_Open(E_GPA,2,E_IO_INPUT);

    DrvGPIO_Open(E_GPA,3,E_IO_OUTPUT);

```

```

DrvGPIO_Open(E_GPA,4,E_IO_OUTPUT);

initUART();
DrvGPIO_ClrBit(E_GPA,2);
DrvGPIO_ClrBit(E_GPA,3);
DrvGPIO_ClrBit(E_GPA,4);
TEXT = '0';
DrvADC_Open(ADC_SINGLE_END, ADC_SINGLE_CYCLE_OP, 0x03,
INTERNAL_HCLK,

while(1)
{
    tampung1 =0;
    bacal = 0;
    arus1 = 0;
for(i=0;i<10;i++)
    {
        DrvADC_StartConvert();
while(DrvADC_IsConversionDone()==FALSE);
bacal = ADC->ADDR[0].RSLT & 0xFFFF;
bacal = bacal*500/4095;
tampung1 = tampung1 + bacal;
DrvSYS_Delay(100);
    }
    tampung1 = tampung1/10;
    arus1 = tampung1;
    tampill[0] = 0x30 + tampung1/100;
    tampung1 = tampung1-tampung1/100*100;
    tampill[1] = ',';
    tampill[2] = 0x30 + tampung1/10;
    tampung1 = tampung1-tampung1/10*10;
    tampill[3] = 0x30 + tampung1;
    tampill[4] = '|';

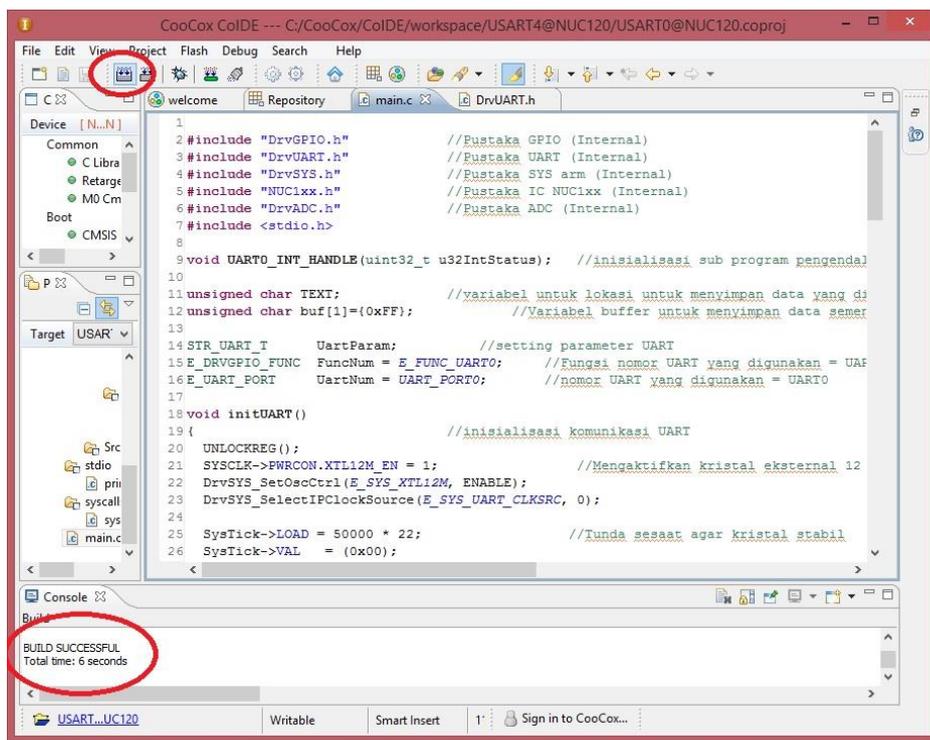
char TEXT1[6] = "ADC:|";
DrvSYS_Delay(5000000);
DrvUART_Write(UartNum,TEXT1,6);

//info sensor arus
if(arus1 < 247){
    char TEXT1[6] = "AMAN|";
    DrvUART_Write(UartNum,TEXT1,6);
}
else if (arus1 >= 247){
    char TEXT1[6] = "BEBAN TINGGI|";
    DrvUART_Write(UartNum,TEXT1,6);
}

//pengukuran ADC1
DrvUART_Write(UartNum,tampill,5);
DrvSYS_Delay(5000000);
}
}

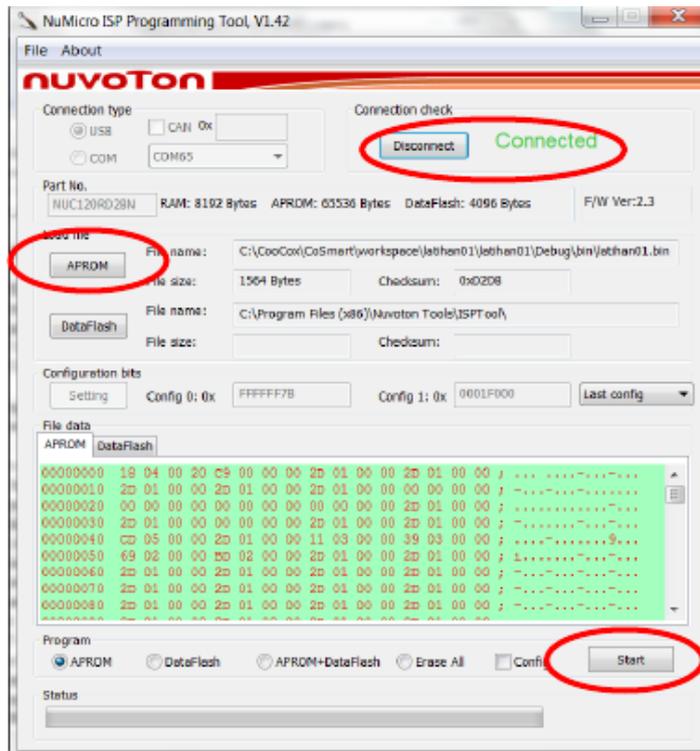
```

Setelah semua program ditulis dalam tab **main.c** pilih dan klik icon **Build** di toolbar untuk melakukan compile program. Pastikan Build Successful, tidak ada error. Seperti pada gambar 3.11



Gambar 3.11 Tampilan Icon Buid Sukses

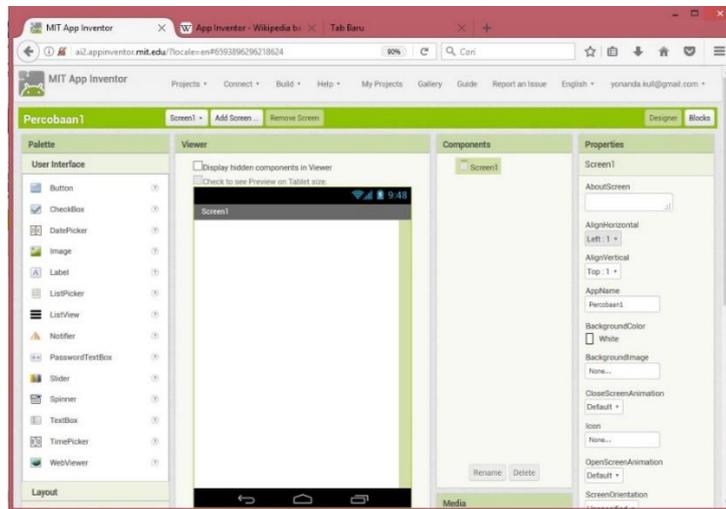
Untuk melakukan flashing ke mikrokontroler nuvoton NUC 120 panggil aplikasi Nuvoton > NuMicro ISP Programming Tool dari Start Menu. Tekan tombol RESET di board DT-ARM NUC120. Klik tombol Connect jika diperlukan sehingga status menjadi Connected (warna hijau). Klik file tombol APROM lalu browse ke **folder project** yang dibuat **\Debug\bin** dan pilih file dengan extensi **“.bin”**. kemudian klik tombol Start untuk memproses download ke hardware, tunggu hingga selesai. Seperti pada gambar 3.12 :



Gambar 3.12 Tampilan flashing ke mikrokontroler NUC 120

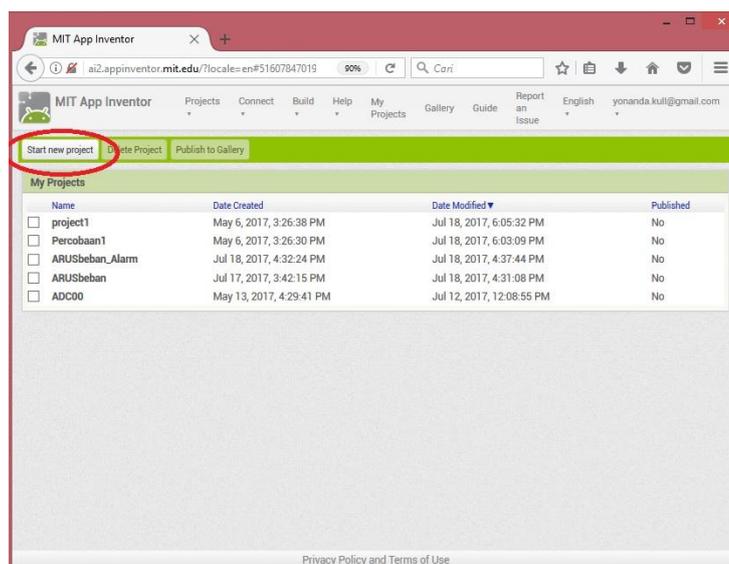
3.3.2 Pemrograman Menggunakan App Inventor

App Inventor adalah aplikasi web sumber terbuka yang awalnya dikembangkan oleh Google, dan saat ini dikelola oleh Massachusetts Institute of Technology (MIT). *App Inventor* memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi perangkat lunak bagi sistem operasi Android. *App Inventor* menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada *Scratch* dan *StarLogo TNG*, yang memungkinkan pengguna untuk men-*drag-and-drop* objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android.



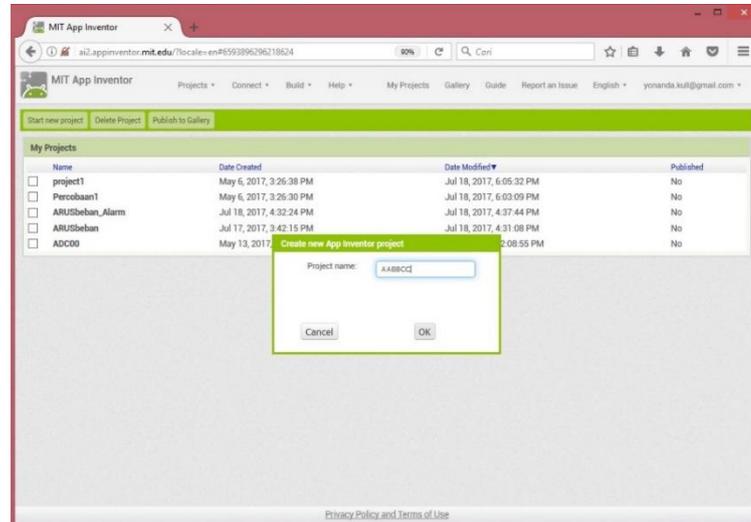
Gambar 3.13. Tampilan *App Inventor*

Untuk memulai pemrograman menggunakan *App Inventor* haruslah memakai web browser, dengan menetik <http://ai2.appinventor.mit.edu/> maka akan muncul jendela login, disini pengguna diharuskan login menggunakan akun google. Selanjutnya akan tampil halaman **My Project**. Pilih pilih pada menu **Start new project** seperti gambar 3.13 :



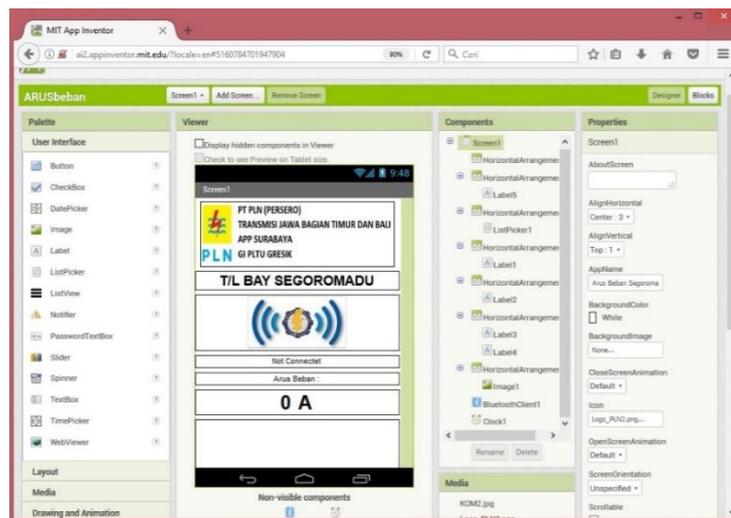
Gambar 3.14. Tampilan *Start new project*

kemudian akan muncul kotak dialog pada gambar 3.15



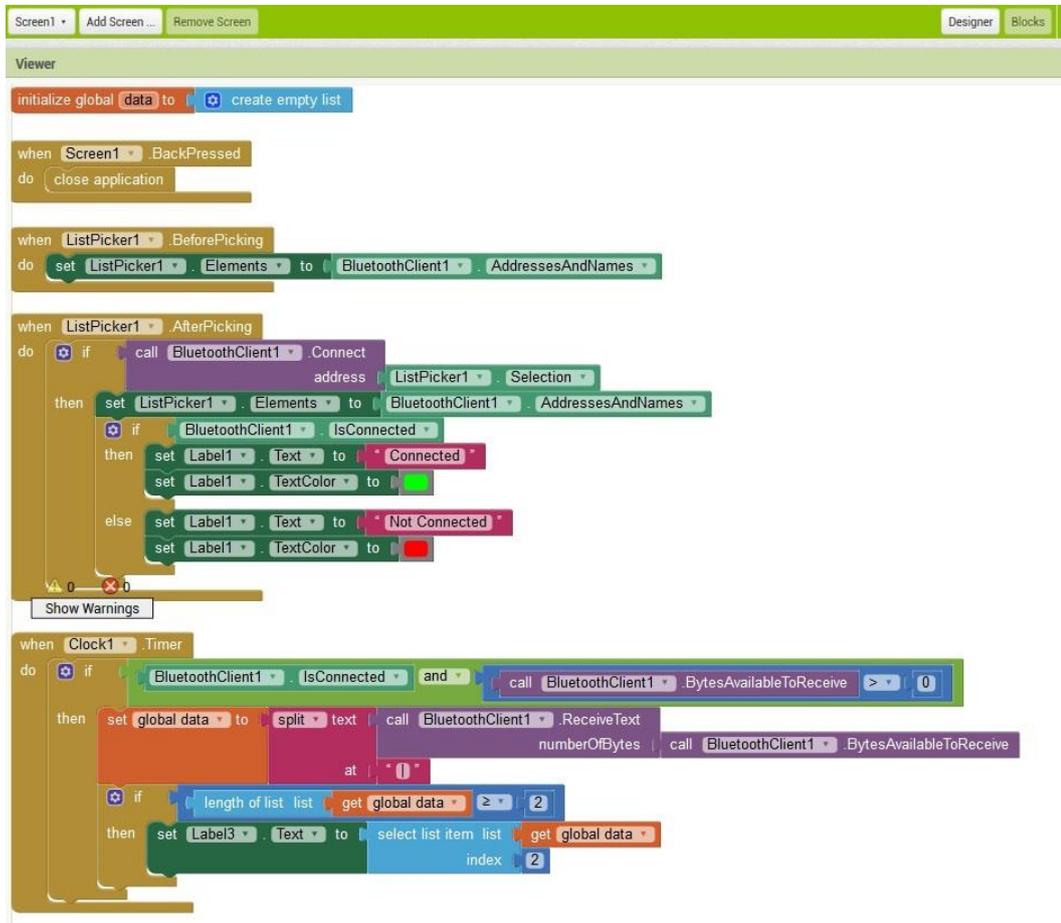
Gambar 3.15. Tampilan jendela pemberian nama project

Isi Project name sesuai dengan nama project yang akan dibuat. Setelah itu tekan **OK** maka akan muncul jendela *Designer*, dimana pada jendela ini desain tampilan pada aplikasi android. Dan untuk tampilan pada jendela *Designer* monitoring arus beban dengan Android seperti pada gambar 3.15



Gambar 3.16. Tampilan *Designer* aplikasi monitoring arus beban dengan Android menggunakan Bluetooth

Sedangkan Untuk **Bloks** programnya adalah seperti gambar 3.17 di bawah ini.



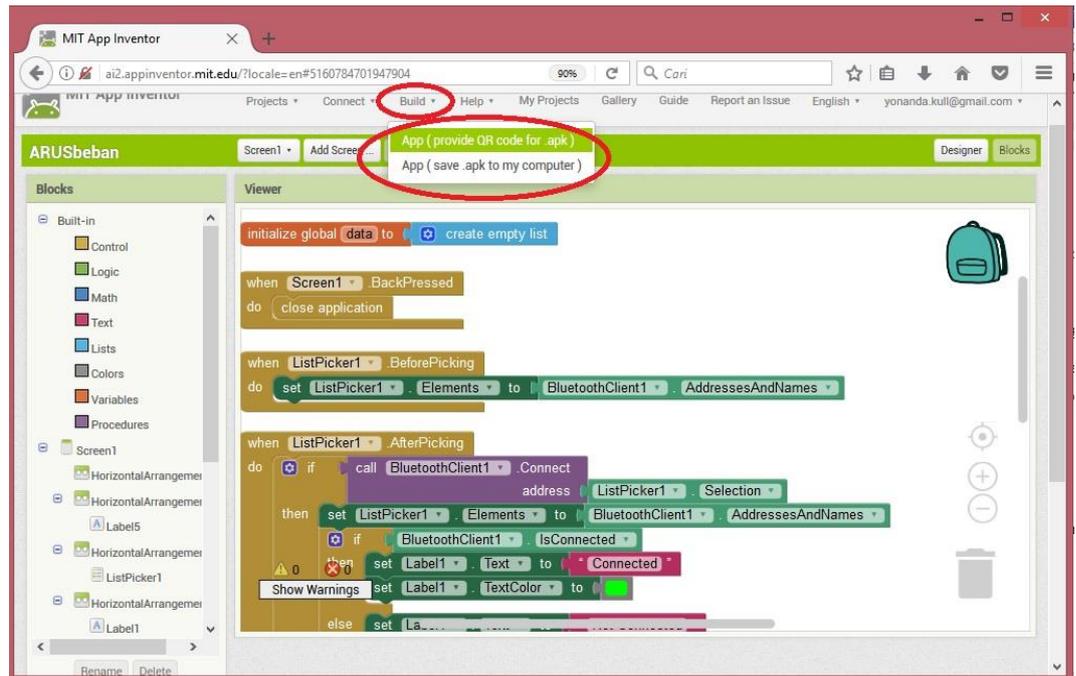
Gambar 3.17. Tampilan **Bloks** aplikasi monitoring arus beban dengan Android menggunakan Bluetooth.

Setelah bloks program aplikasi telah tersusun dengan benar dengan memastikan tidak ada kesalahan maupun peringatan seperti pada gambar 3.18 :



Gambar 3.18. Tidak ada peringatan maupun kesalahan pada program.

Untuk melakukan *compile* program yang telah dibuat terdapat dua cara untuk download program ke android. Cara pertama adalah dengan mode *scan QR code*, klik menu **Build>App (provide QR code for .apk)** seperti pada gambar 3.19 :



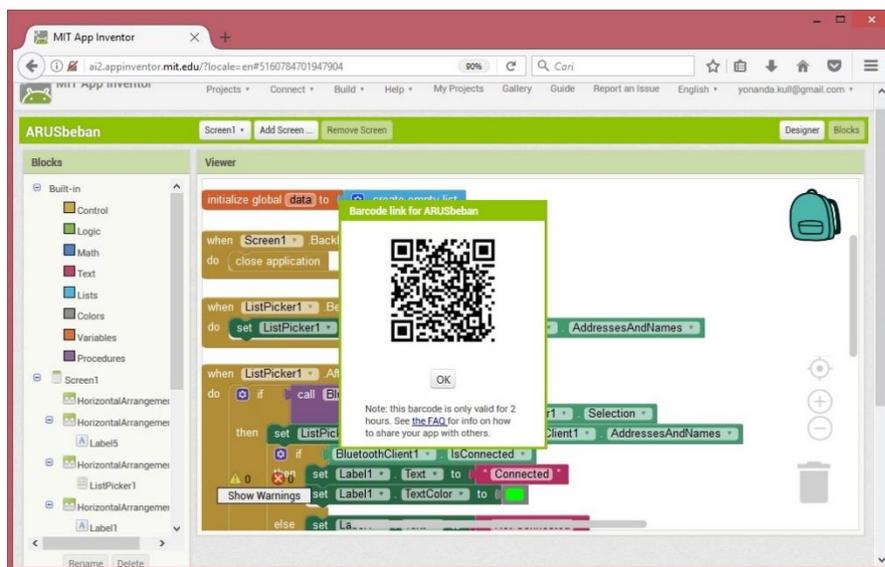
Gambar 3.19. Build QR code .apk

Selanjutnya akan muncul tampilan proses build .apk seperti pada gambar 3.20 di bawah ini:



Gambar 3.20. Proses Build .apk

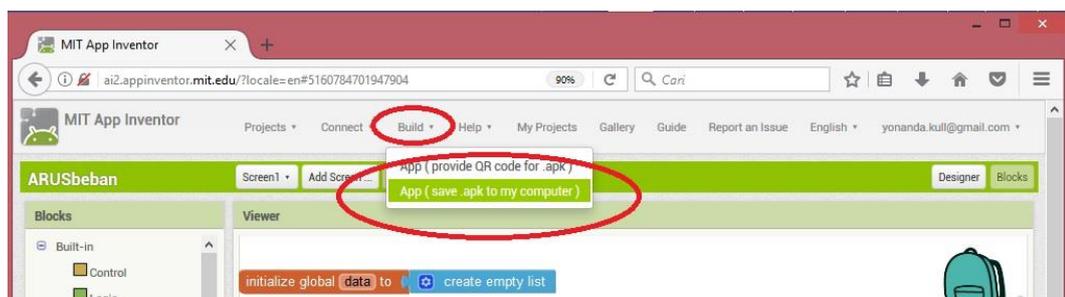
Tunggu hingga proses build . apk selesai, proses ini akan memakan waktu cukup lama tergantung oleh kecepatan koneksi internet. Setelah build .apk selesai maka akan muncul tampilan *QR code* seperti pada gambar 3.21 :



Gambar 3.21. QR Code build .apk

Selanjutnya buka aplikasi pada android yang bisa mengakses QR code dan lakukan scanning pada QR code lalu pada android akan muncul alamat untuk mendownload dan install aplikasi yang telah dibuat.

Cara yang ke dua adalah dengan mendownload file dengan format .apk langsung dari PC/Laptop, klik menu **Build>App (save .apk to my computer)** seperti pada gambar 3.22 :



Gambar 3.22. Build .apk

Selanjutnya akan muncul tampilan proses build .apk seperti pada gambar

3.23 di bawah ini:



Gambar 3.23. Proses Build .apk

Hingga proses build apk selesai, PC/Laptop akan otomatis mendownload file ber format .apk. Setelah file ber format .apk tersebut berhasil di donwload pada PC/Laptop, kirim data ber format .apk tersebut ke Android melalui kabel data dan lakukan instalasi aplikasi langsung dari Android. Aplikasi yang berhasil terinstal tersebut akan muncul pada Android seperti pada gambar 3.24 :



Gambar 3.24. Aplikasi monitoring Android