PROGRESS

Untuk membuat LED Blink langkah-langkahnya seperti berikut :

Beberapa hal yang perlu di sediakan.  
\* Driver USB ST-Link                                              
\* Software CooCox CoIDE                                      
\* GNU Tools for ARM Embedded Processors        
\* Kabel data USB  mini tipe B  
\* Module STM32F4 Discovery.

1. Download kedua file program diatas.  
2. Install Software CooCox CoIDE dan Driver USB ST-Link.  
3. Install GNU Tools for ARM Embedded Processors.  
4. Setting tool chain pada CoIDE, untuk seting toolchain bisa lakukan langkah dibawah ini :

a. Jalankan program CoIDE, klik “Poject” -> “Select ToolChain Path” pada menu utama.   
b. Klik tombol “Browse”, cari folder tempat penyimpanan file “arm-none-eabi-gcc.exe”, folder  ini terdapat pada folder instalasi GNU Tools for ARM Embedded Processors.    
c. Klik tombol “OK”, akan muncul peringatan jika folder yang dipilih tidak benar.

5. Pembuatan program Bliking LED

a. Klik “Project” -> “New Project”   
b. Isikan nama Project -> klik tombol “Next”    
c. Pilih tombol “Chip” -> klik tombol “Next”    
d. Pilih jenis processor, dalam hal ini STM32F407VGT -> klik tombol “Finish”    
e. Pada tab “Repository” pilih “GPIO”, CoIDE akan menambahkan file-file yang dibutuhkan untuk mengakses PORT I/O processor pada project kita.    
f. Double klik “main.c” pada file list project, biasanya terdapat pada sisi kiri-bawah program CoIDE, maka CoIDE akan membuka file “main.c” yang merupakan file utama project kita.    
g. Menuliskan program, untuk listing program Blinking LED bisa dilihat dibawah ini.

#include "stm32f4xx\_gpio.h" // file header untuk akses PORT I/O

#include "stm32f4xx.h" // file header basic system STM32F4

#include "stm32f4xx\_rcc.h"

#define LED\_BLUE\_ON   GPIOD->BSRRL = GPIO\_Pin\_15;  // led biru on

#define LED\_BLUE\_OFF  GPIOD->BSRRH = GPIO\_Pin\_15;  // led biru off

#define LED\_RED\_ON   GPIOD->BSRRL = GPIO\_Pin\_14;  // dst

#define LED\_RED\_OFF  GPIOD->BSRRH = GPIO\_Pin\_14;

#define LED\_ORANGE\_ON   GPIOD->BSRRL = GPIO\_Pin\_13;

#define LED\_ORANGE\_OFF  GPIOD->BSRRH = GPIO\_Pin\_13;

#define LED\_GREEN\_ON   GPIOD->BSRRL = GPIO\_Pin\_12;

#define LED\_GREEN\_OFF  GPIOD->BSRRH = GPIO\_Pin\_12;

void inisialisasi() // Inisialisasi port D, port tempat LED berada

{

    GPIO\_InitTypeDef  GPIO\_InitStructure;

    RCC\_AHB1PeriphClockCmd(RCC\_AHB1Periph\_GPIOD, ENABLE);

    GPIO\_InitStructure.GPIO\_Pin = GPIO\_Pin\_12 | GPIO\_Pin\_13 |GPIO\_Pin\_14|GPIO\_Pin\_15;

    GPIO\_InitStructure.GPIO\_Mode = GPIO\_Mode\_OUT;

    GPIO\_InitStructure.GPIO\_OType = GPIO\_OType\_PP;

    GPIO\_InitStructure.GPIO\_Speed = GPIO\_Speed\_50MHz;

    GPIO\_Init(GPIOD, &GPIO\_InitStructure);

}

void delay(uint32\_t ulang) // procedure delay

{

while (ulang > 0)

{ulang--;}

}

int main(void)

{  inisialisasi();        //memanggil fungsi inisialisasi

   while(1)               //proses berulang terus menerus

    {   LED\_BLUE\_ON;

        LED\_RED\_OFF;

        delay(1000000);

        LED\_BLUE\_OFF;

        LED\_GREEN\_ON;

        delay(1000000);

        LED\_GREEN\_OFF;

        LED\_ORANGE\_ON;

        delay(1000000);

        LED\_ORANGE\_OFF;

        LED\_RED\_ON;

        delay(1000000);

    }

}

6. Compile dan Download program ke module.

a. Untuk kompilasi klik tombol “Build” atau tekan tombol “F7” pada key board. bisa dilihat apakah ada kesalahan pada listing program pada “Console”, sisi bawah program.

b. Untuk download program ke modul hubungan modul ke komputer menggunakan kabel data USB kemudian klik tombol “Download Code To Flash”.

Jika semua berjalan dengan benar maka 4 LED pada modul akan menyala bergantian.