RL78/G12

Remotely Controllable Button Pusher

Introduction

This application note describes an example to control the W-Fi module ESP-WROOM-02 by using RL78/G12 and control the button pusher via a network.

The application note "RL78 / G10 Wi-Fi module (ESP-WROOM-02) control sample software for TCP/IP Slave Transmission/Reception" is used to control the Wi-Fi module.

Target Device

RL78/G12

When applying the sample program covered in this application note to another microcomputer, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

Contents

1. Specifications

Figure 1.1 shows the system configuration. In this application note, the button pusher is controlled via a network using the Wi-Fi module ESP-WROOM-02.

When the button pusher is powered on, the RL78/G12 controls the Wi-Fi module and obtains an IP address from the access point. After that, it notifies the TCP server (PC, smartphone, etc.) that the button pusher is connected to the network, and the button pusher waits for reception from the TCP server.

When receiving data from the TCP server, the RL78/G12 analyzes the received data and executes the operation according to the received data. Port output and the PWM output of the Timer Array Unit are used for DC motor control.

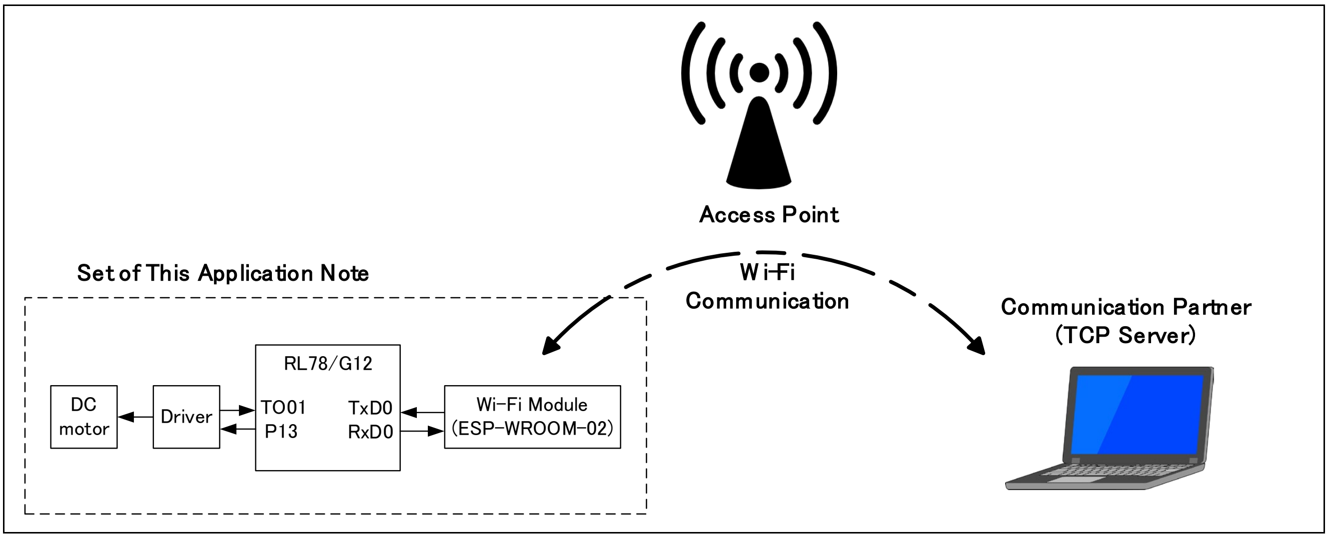


Figure 1.1 System Configuration

1.1 Control of DC motor

Use a motor driver to control a DC motor. RL78/G12 controls Port output and PWM waveform as input signals for the motor driver.

Table 1.1 shows the relationship between RL78/G12 and DC motor status.

Table 1.1 Relationship between RL78/G12 and DC motor status

RL78/G12		State of DC motor
P13 output	**TO01 (PWM output)**	
0	1	Normally rotated
0	0	Stop
1	0	Stop
1	1	Inversely rotated

Figure 1.2 shows how to control the DC motor. As process of pushing the button, normal rotation, stop, and inverse rotation are executed. Press the button during normal rotation and release the button during inverse rotation. The rotation time of the DC motor can be adjusted by software.

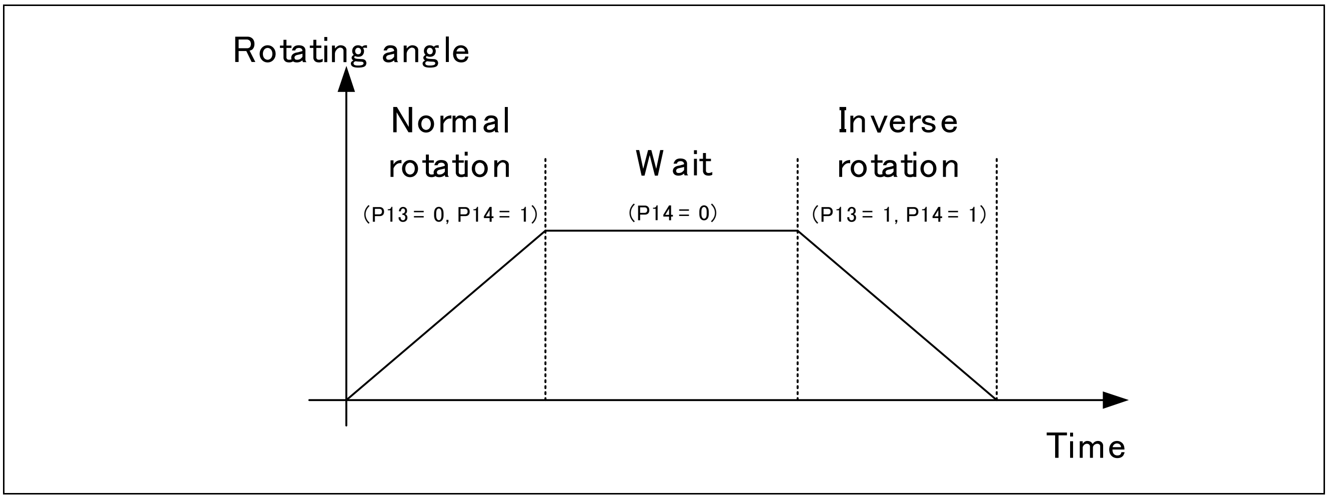


Figure 1.2 How to control DC motor

1.2 Format of the communication packet

Table 1.2 shows the data sent from the TCP server. The data is composed of bytes of ASCII codes.

Figure 1.2 Data sent from the TCP server

Data	Process
"ON&OFF\n"	Normal and inverse operation of DC motor

2. Conditions of Operation Confirmation Test

The sample code with this application note runs properly under the conditions below.

Table 2.1 Operation Confirmation Conditions

Items	Contents
MCU	RL78/G12（R5F1026A）
Operating frequencies	High-speed on-chip oscillator clock (fIH): 24MHz CPU/peripheral hardware clock: 24 MHz
Operating voltage	3.3V LVD operations (VLVD): Rising edge TYP. 2.81V (2.76V～2.87V) Falling edge TYP. 2.75V (2.70V～2.81V)
Integrated development environment (CS+)	CS+ for CC V8.10.00 from Renesas Electronics Corp.
C compiler (CS+)	CC-RL V1.12.01 from Renesas Electronics Corp.
Integrated development environment (e2studio)	e2 studio 2023-01 (23.1.0) from Renesas Electronics Corp.
C compiler (e2studio)	CC-RL V1.12.01 from Renesas Electronics Corp.
Integrated development environment (IAR)	IAR Embedded Workbench for Renesas RL78 V4.21.3 from IAR Systems
C compiler (IAR)	IAR C/C++ Compiler for Renesas RL78 V4.21.3.2447from IAR Systems
Wi-Fi Module	ESP-WROOM-02 (ESP8266EX) from Espressif Systems AT version:1.7.3.0 SDK version:3.0.3

3. Hardware

3.1 Hardware configuration

Figure 3.1 shows an example of the hardware configuration used in this application note.

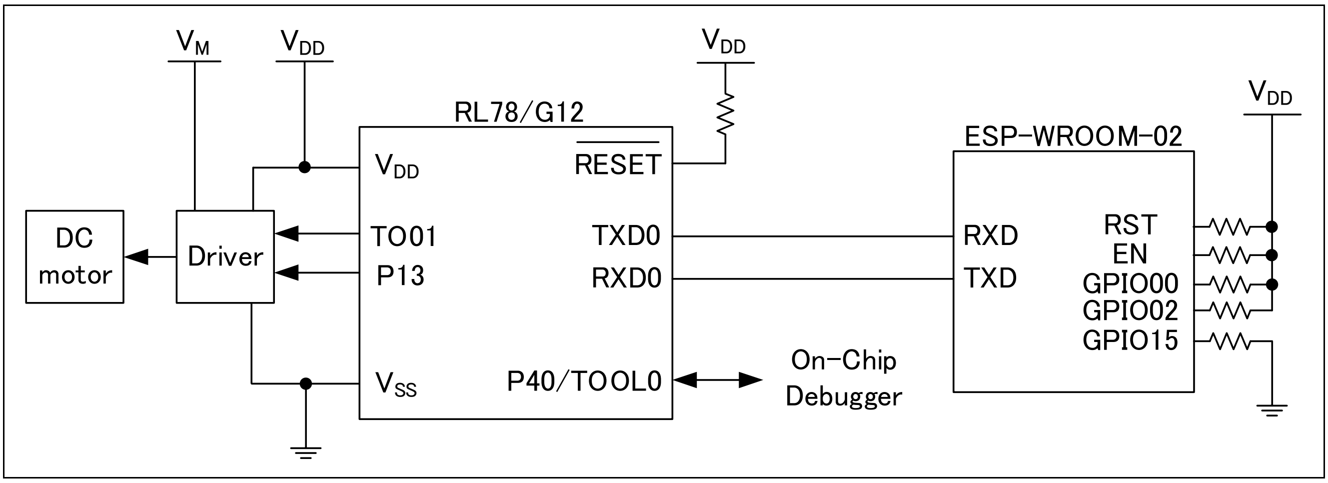


Figure 3.1 Hardware Configuration

Note 1: This simplified circuit diagram was created to show an overview of connections only.

When actually designing your circuit, make sure the design includes sufficient pin processing and meets electrical characteristic requirements. (Connect each input-only port to V_{DD} or V_{SS} through a resistor.)

3.2 Used Pins

Table 3.1 shows list of used Pins and assigned functions.

Table 3.1 List of Pins and Functions

Pin Name	Input/Output	Function
P12/TXD0	Output	Serial data transmission (UART)
P11/RXD0	Input	Serial data reception (UART)
P13	Output	Control rotate direction of DC motor
P14/TO01	Output	Control ON/OFF of DC motor (PWM)

4. Software

4.1 Overview of the Sample Software

In this application note, RL78/G12 analyzing the data sent from the TCP server and execute operations according to the received data. The DC motor is controlled using the INTTM00 interrupt that occurs every 40us.

(1) Initial settings for the RL78/G12.

(2) Initial settings for the Wi-Fi module.

(3) Connect to the access point and get an IP address.

(4) Connect to the TCP server and notify that the button press device is connected to the network.

(5) RL78/G12 waits for data sent from the TCP server.

(6) RL78/G12 stores the received data in RAM and analyzes the data.

(7) RL78/G12 operates the DC motor according to the received data.

(8) RL78/G12 repeats (5) to (7).

4.2 Option Byte Settings

Table 4.1 lists the option byte settings.

Table 4.1 Option Byte Settings

Address	Setting Value	Contents
000C0H	11101111B	Operation of Watchdog timer is stopped （counting is stopped after reset.）
000C1H	01111111B	LVD operations (VLVD): Rising edge TYP. 2.81V (2.76V～2.87V) Falling edge TYP. 2.75V (2.70V～2.81V)
000C2H	11100000B	High-speed on-chip oscillator clock: 24 MHz
000C3H	10000101B	On-chip debugging enabled

4.3 Global Variables

Table 4.2 lists the global variables used in this application note.

Table 4.2 Global Variables

Type	Name	Contents	Functions used in
uint16_t	g_duty	Duty of PWM waveform	main()
uint16_t	g_on_time	Rotating time of the DC motor	main()
uint16_t	g_wait_time	Wait time of the DC motor	main()

4.4 Constants

Table 4.3 lists the constants used in this application note.

Table 4.3 Global Variables

Constant Name	Value	Contents
FCLK_MHZ	24	CPU clock frequency (fCLK) [MHz]

4.5 Functions

Table 4.4 lists functions used in this application note.

Please refer to the application note “RL78/G10 Wi-Fi module (ESP-WROOM-02) control sample software for TCP/IP Slave Transmission/Reception” for specifications of Wi-Fi module controlling functions.

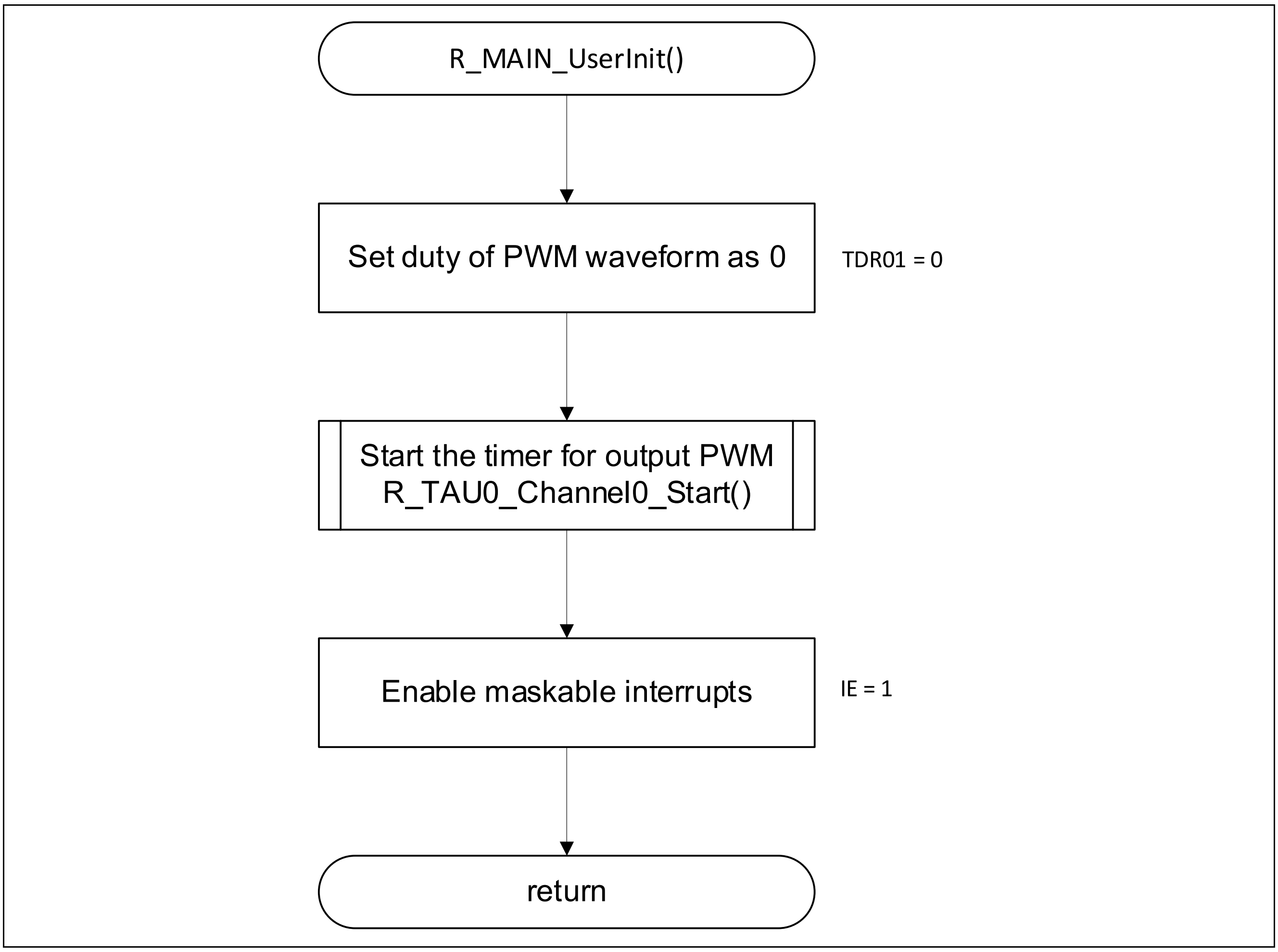
Table 4.4 Functions

Function Name	Outline
main	Main function
R_MAIN_UserInit	User initialization function
R_start_motor	Start to drive the DC motor
R_wait_100ms	Wait for 100ms
R_wait_1s	Wait for 1s

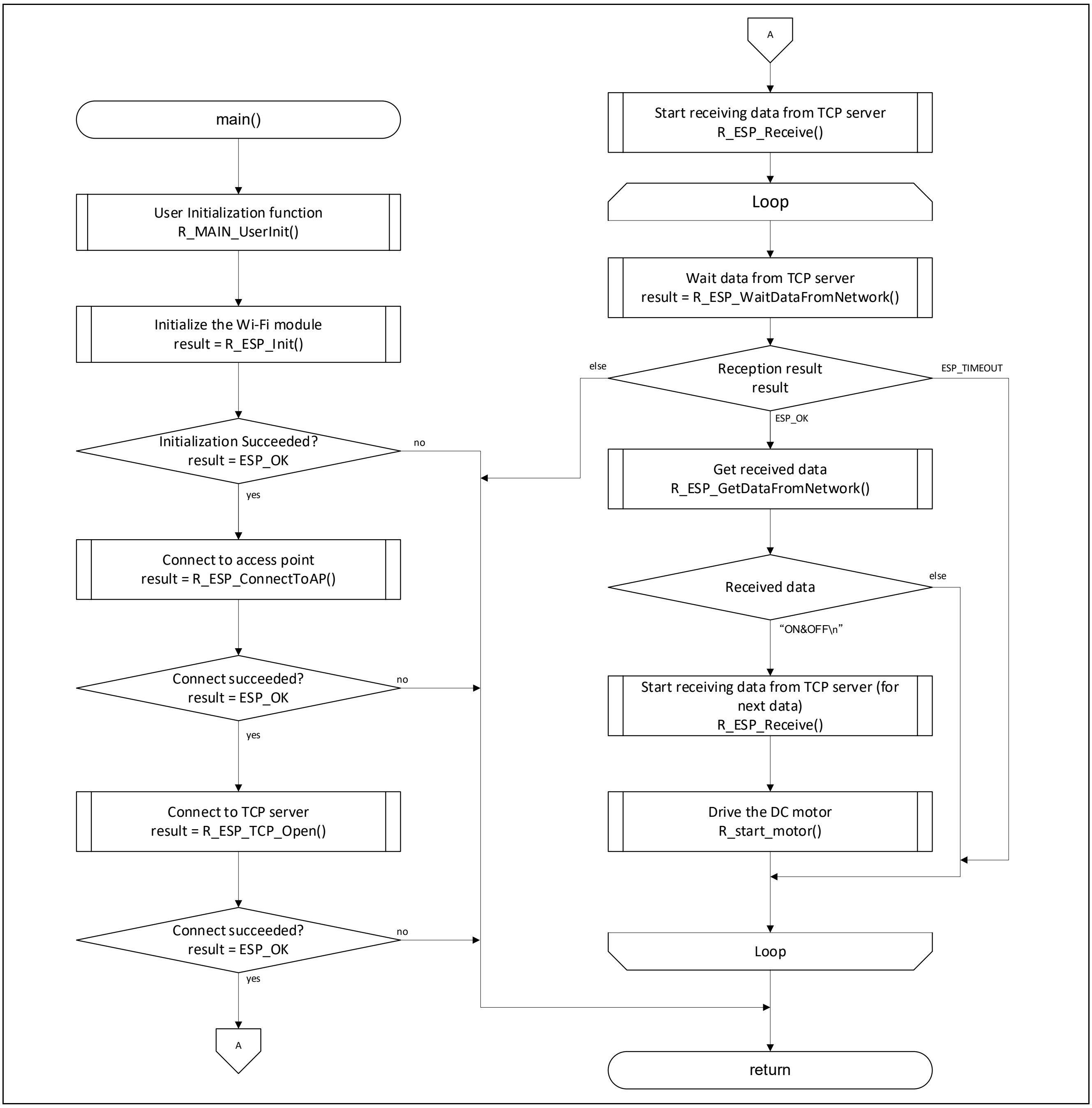
4.6 Function Specifications

The following describes detailed specifications and flowchart of the functions used in this application note.

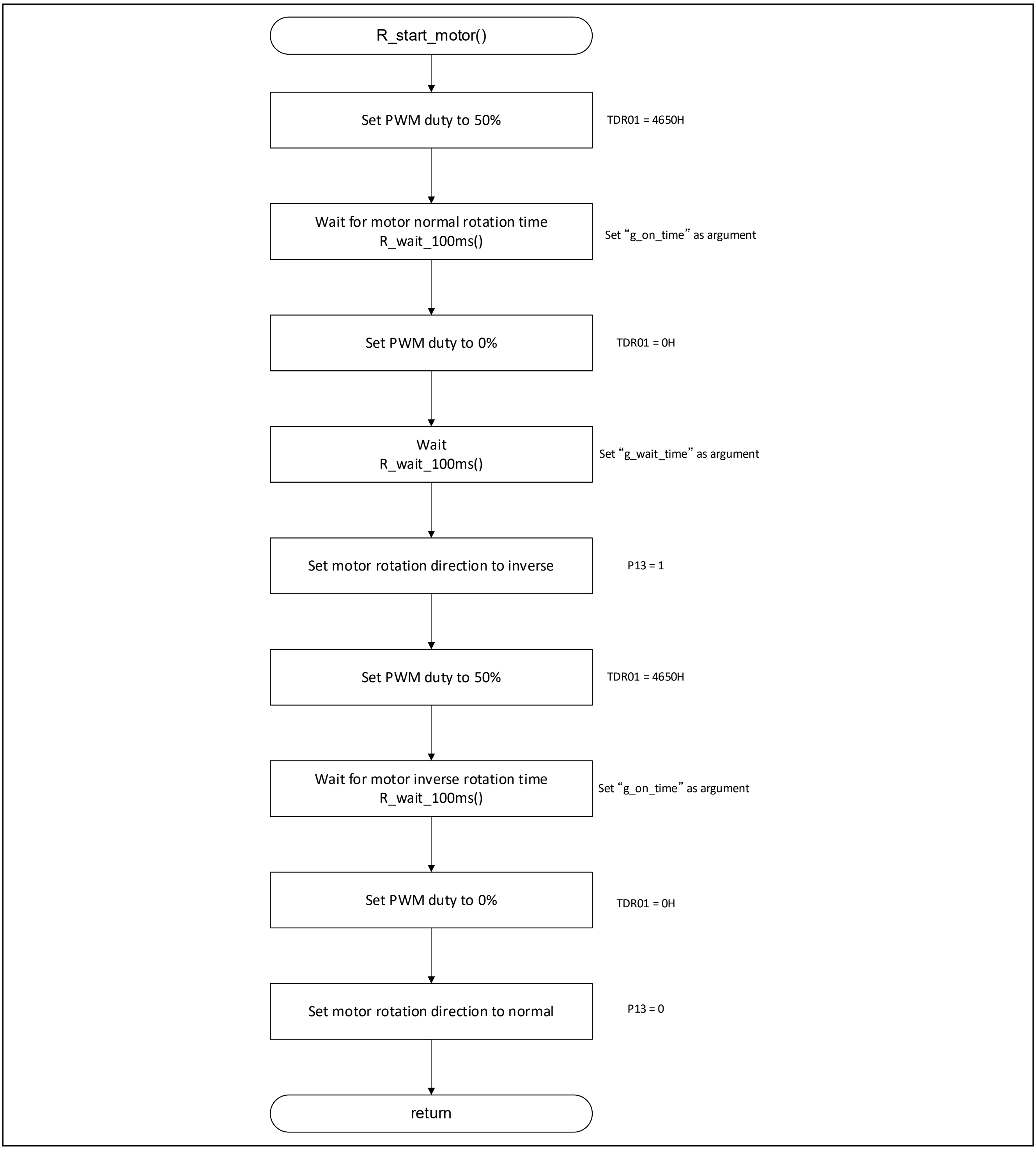
[Function Name]	R_MAIN_UserInit
Outline	User initial setting
Header	None
Declaration	static void R_MAIN_UserInit(void);
Description	User initialization function
Arguments	None None
Return value	None
Remarks	None



[Function Name]	main
Outline	main function
Header	—
Declaration	—
Description	After executing the main user initialization function, control the Wi-Fi module and connect to the TCP server as a client. After that, RL78/G12 waits in the communication waiting state. When receiving data from the Wi-Fi module, it analyzes the received data and controls the DC motor by controlling the motor driver according to the received data. If a Wi-Fi module error occurs, exit the program.
Arguments	None
Return value	None
Remarks	None

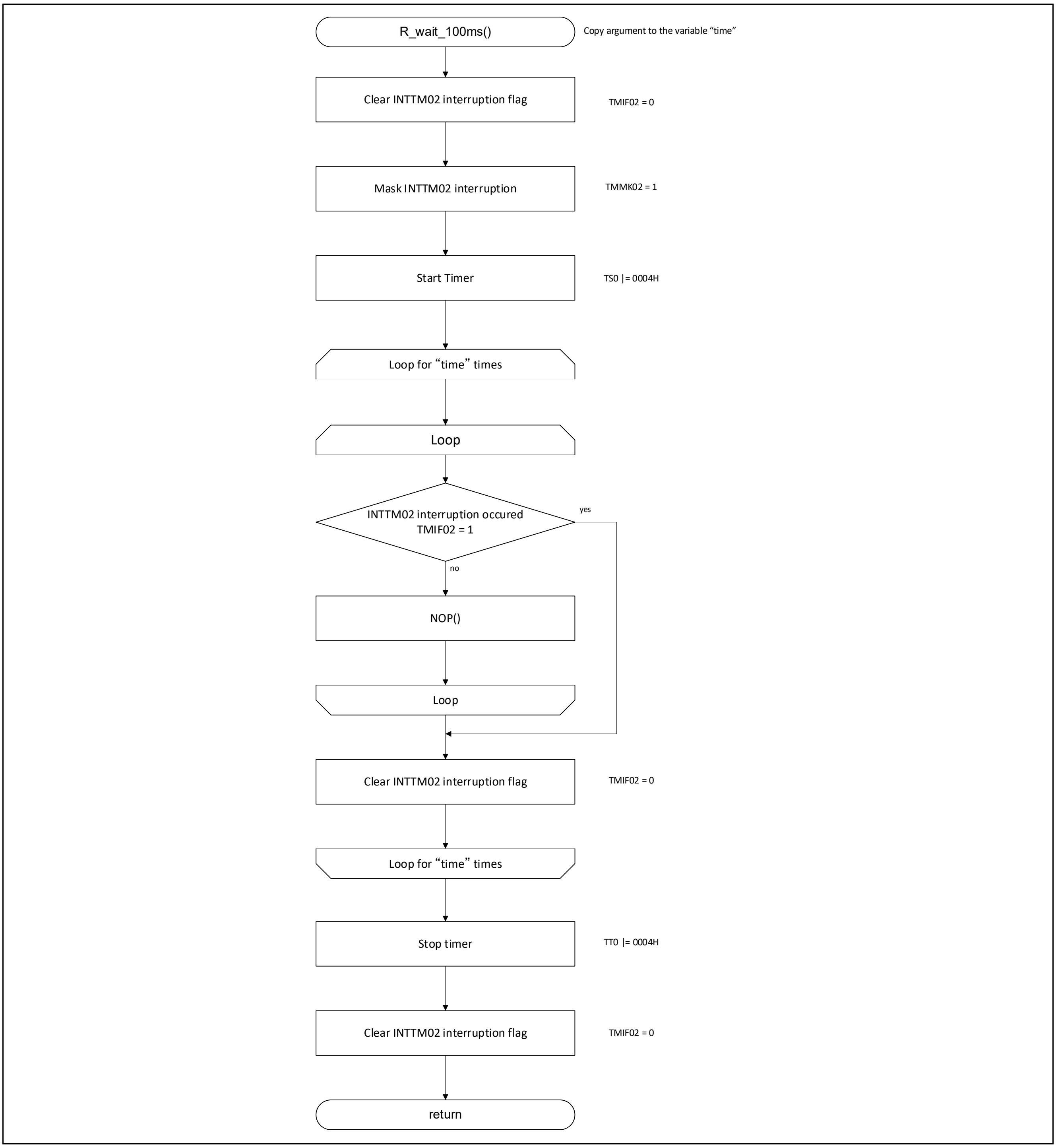


[Function Name]	**R_start_motor**
Outline	Start to drive the DC motor
Header	None
Declaration	void R_TAU0_Channel0_Start(void);
Description	Output signals to control the DC motor
Arguments	None
Return value	None
Remarks	None

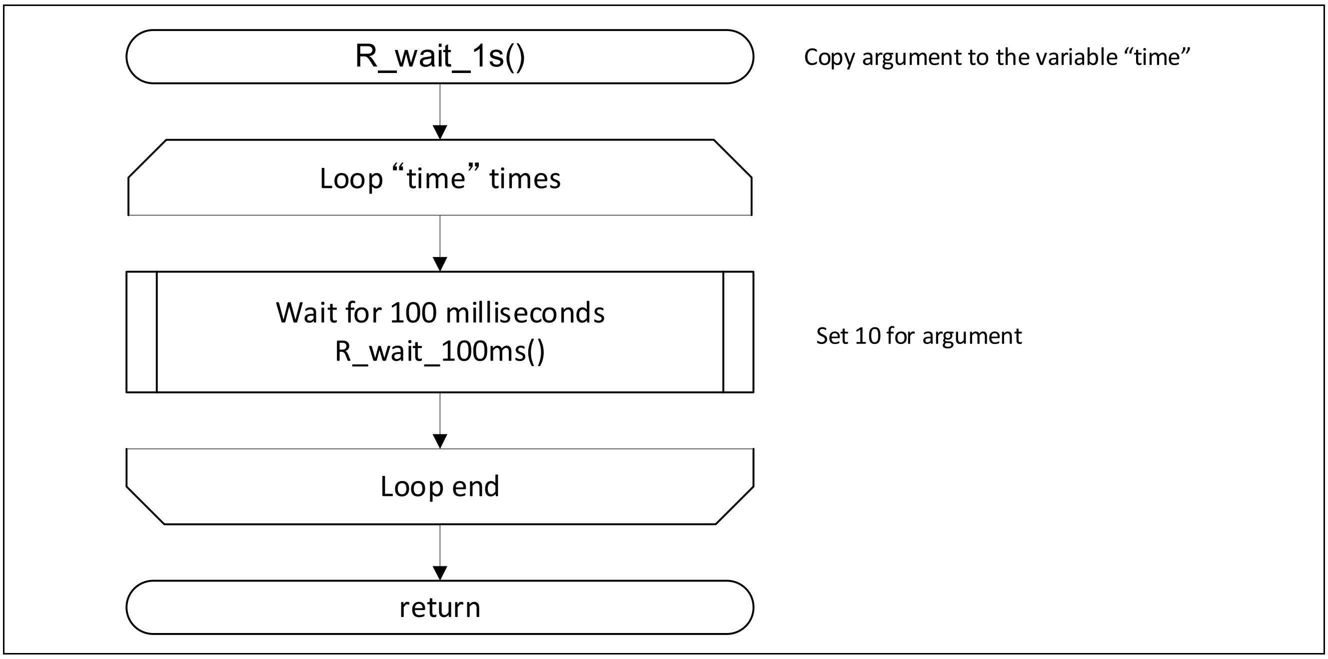


[Function Name] R_wait_100ms

Outline	wait for 100ms
Header	None
Declaration	void R_wait_100ms(uint16_t time);
Description	Waits for 100ms. Executes 100ms wait for the number of times designated by the argument.
Arguments	uint16_t time Wait time [100ms]
Return value	None
Remarks	None



[Function Name]	**R_wait_1s**	
Outline	Wait for 1 second	
Header	None	
Declaration	void R_wait_1s(uint8_t time);	
Description	Waits for 1 second. Executes 1s wait for the number of times designated by the argument.	
Arguments	uint8_t time	Wait time [s]
Return value	None	
Remarks	None	



5. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

Referenced Application Note:
"RL78 / G10 Wi-Fi module (ESP-WROOM-02) control sample software for TCP/IP Slave Transmission/Reception"

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 9, 2020	—	First Edition
1.10	June. 24. 22	5	Operation check condition is updated.
1.11	Aug. 4. 23	5	Operation check condition is updated.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.