

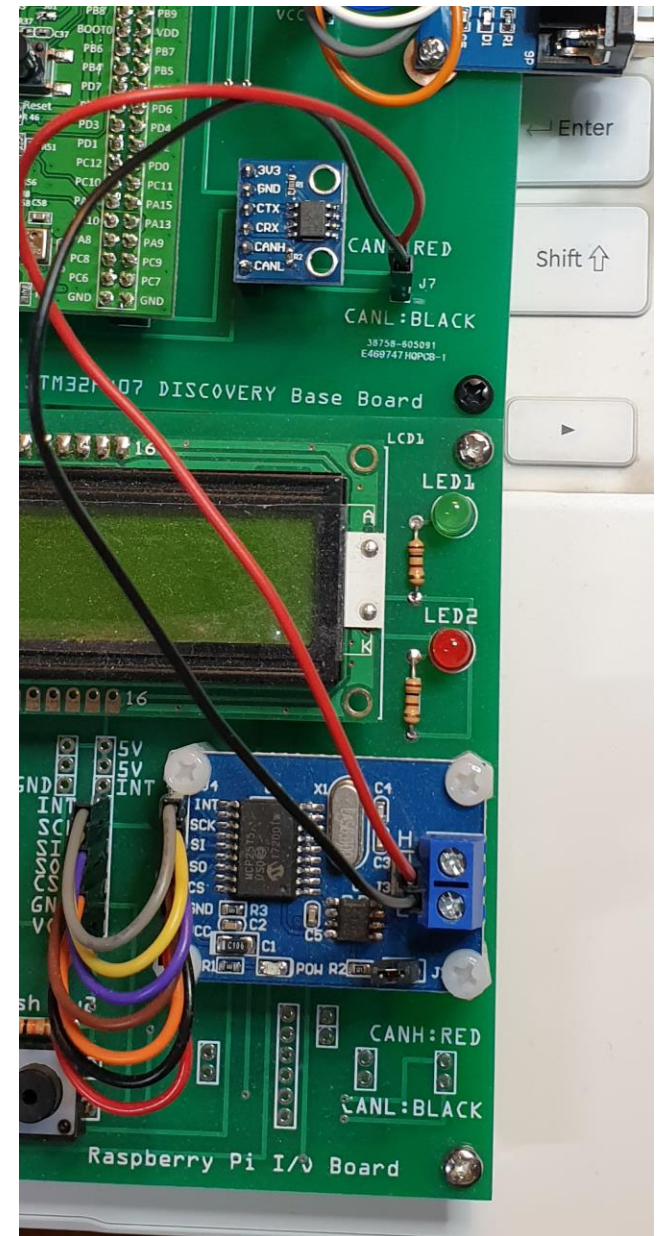
Lab Cortex-M4

CAN Communication



Before you begin

- Connect CAN terminals of STM32F407 Discovery board and MCP2515 board connected to the Raspberry Pi
- CANH-CANH, CANL-CANL



New STM32 project

IDE STM32 Project

Setup STM32 project

Project

Project Name:

Use default location

Location:

Options

Targeted Language

C C++

Targeted Binary Type

Executable Static Library

Targeted Project Type

STM32Cube Empty

Pinout & Configuration

- CAN1: Activated
- USART2:
Asynchronous
- USART3:
Asynchronous

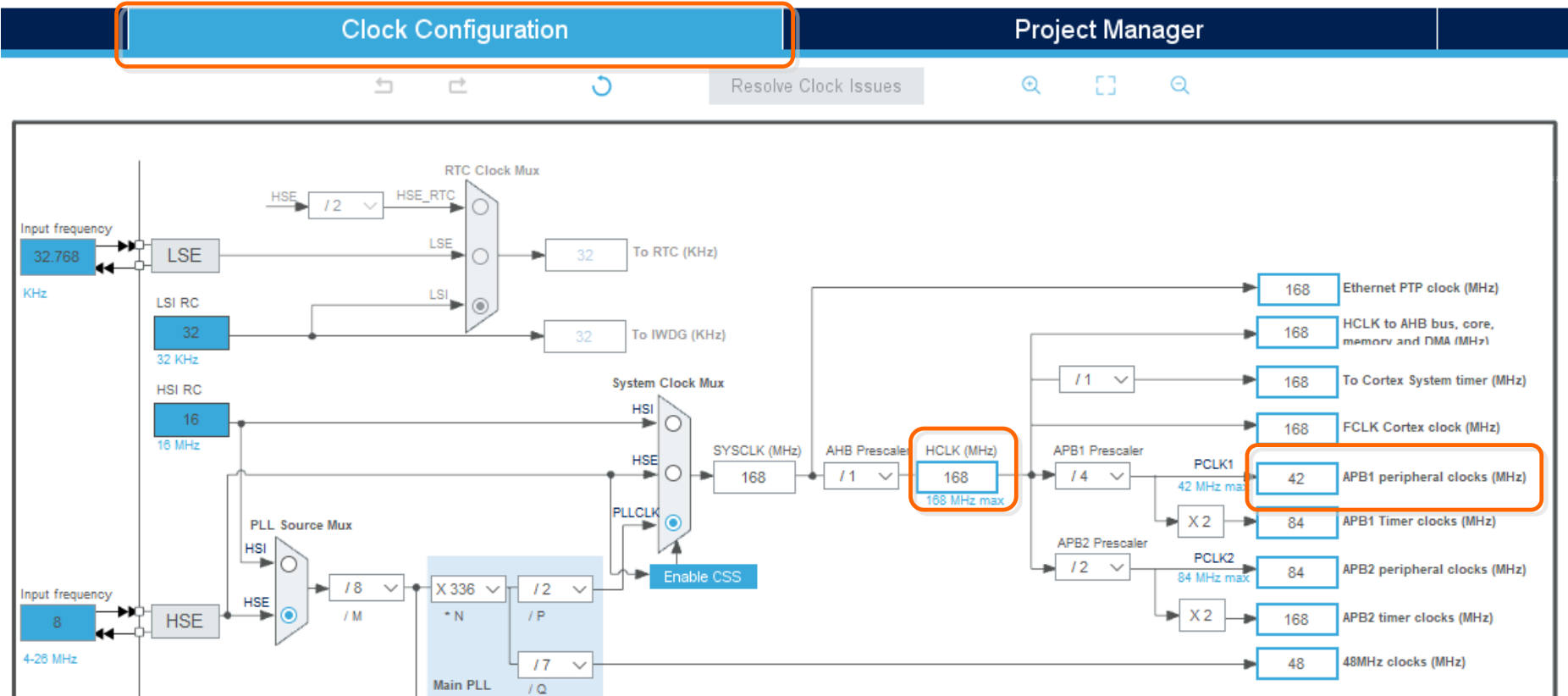
The screenshot displays the 'Pinout & Configuration' software interface. The main window is titled 'Pinout & Configuration' and has a 'Clock Config' button in the top right corner. Below the title bar, there is a search bar and a 'Software Pack' dropdown menu. The main content area is titled 'CAN1 Mode and Configuration' and shows a 'Mode' section with a checked checkbox for 'Activated'. At the bottom of the interface, there is a 'Configuration' section with a 'Reset Configuration' button.

Categories: A->Z

- ✓ CAN1
- CAN2
- ⊗ ETH
- FSMC
- ✓ I2C1
- ⊗ I2C2
- ⚠ I2C3
- ⊗ SDIO
- ✓ SPI1
- SPI2
- SPI3
- ⊗ UART4
- ⊗ UART5
- ⊗ USART1
- ✓ USART2
- ✓ USART3

Clock Configuration

- Select Clock Configuration Tab
- Check 42MHz for APB1 peripheral clocks



CAN Parameters

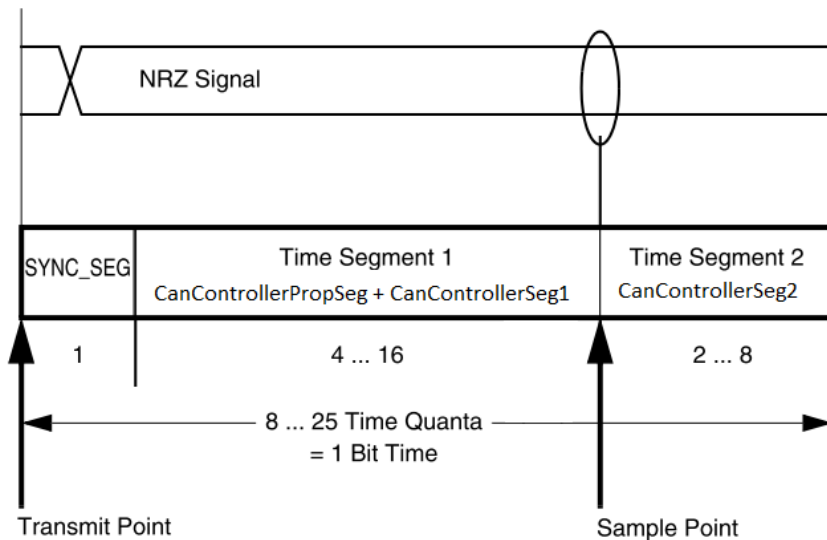
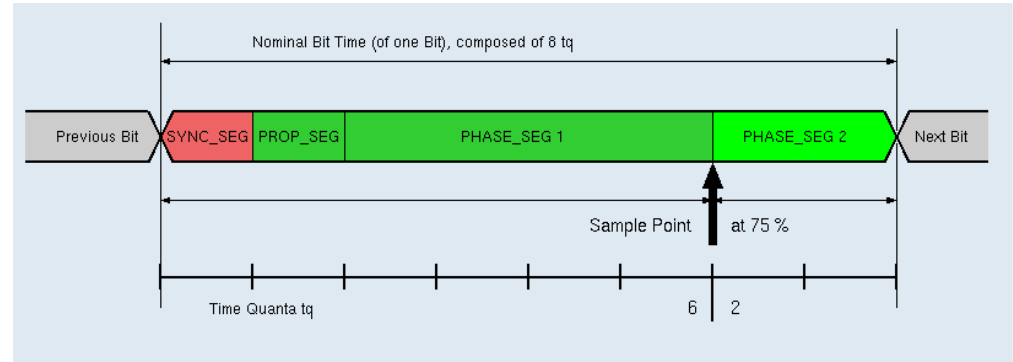
- Select Pinout & Configuration Tab
- Click CAN1 and select Parameter Settings
- Change Time Quanta to 9 Times and 4 Times
- Change Prescaler to 6

The screenshot displays the STM32CubeMX configuration tool. The 'Pinout & Configuration' tab is selected, and the 'Parameter Settings' sub-tab is active. The 'Mode' section shows 'Activated' checked. The 'Configuration' section shows 'Parameter Settings' selected. The 'Bit Timings Parameters' section shows the following values:

Parameter	Value
Prescaler (for Time Quantum)	6
Time Quantum	142.85714285714286 ns
* Time Quanta in Bit Segment 1	9 Times
* Time Quanta in Bit Segment 2	4 Times
* Time for one Bit	1999.99 ns
* Baud Rate	500000 bit/s
ReSynchronization Jump Width	1 Time

CAN bit timing

- $42\text{MHz}/6=7\text{MHz}$
- $1/7\text{MHz}=142.851743\text{ nsec}$



CAN Interrupt Setting

- Select NVIC Settings
- Check CAN1 RX0 interrupts

- ✓ CAN1
- CAN2
- ⊗ ETH
- FSMC
- ✓ I2C1
- ⊗ I2C2
- ⚠ I2C3
- ⊗ SDIO
- ✓ SPI1
- SPI2
- SPI3

Configuration

Reset Configuration

✓ User Constants ✓ NVIC Settings ✓ GPIO Settings

✓ Parameter Settings

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
CAN1 TX interrupts	<input type="checkbox"/>	0	0
CAN1 RX0 interrupts	<input checked="" type="checkbox"/>	0	0
CAN1 RX1 interrupt	<input type="checkbox"/>	0	0
CAN1 SCE interrupt	<input type="checkbox"/>	0	0

main.c(1)

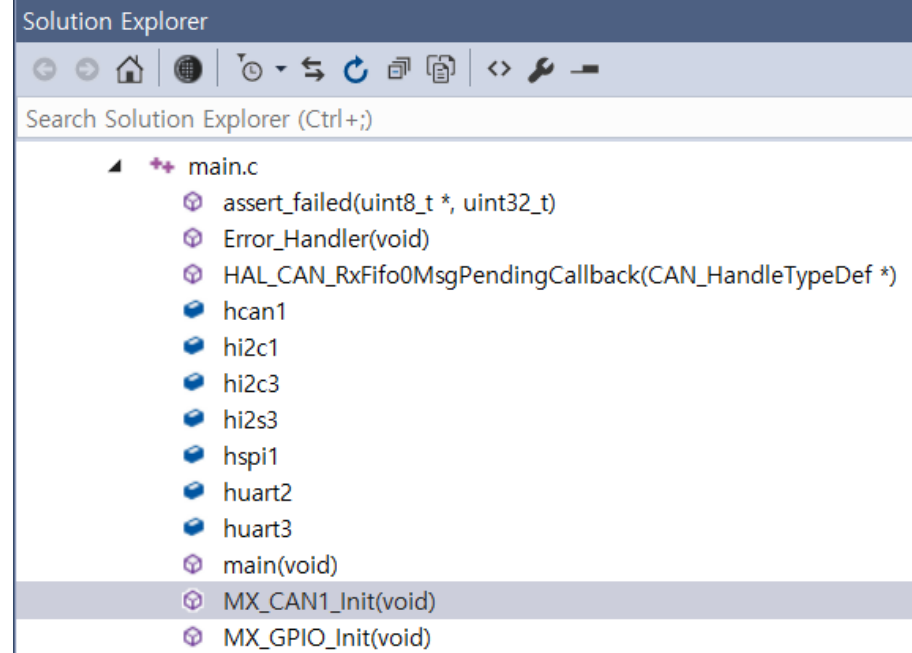
```
/* USER CODE BEGIN PV */
CAN_TxHeaderTypeDef TxHeader;
CAN_RxHeaderTypeDef RxHeader;
uint8_t TxData[8];
uint8_t RxData[8];
uint32_t TxMailbox;
/* USER CODE END PV */

/* USER CODE BEGIN 2 */
/* Test CAN data transmission */
TxData[0] = 0x00;
//while(1) {
    TxData[0]++;
    if (HAL_CAN_AddTxMessage(&hcan1, &TxHeader, TxData, &TxMailbox) != HAL_OK) {
        /* Transmission request Error */
        Error_Handler();
    }
    HAL_Delay(1000);
}
/* USER CODE END 2 */
```

main.c(2)

```
static void MX_CAN1_Init(void)
{

/* USER CODE BEGIN CAN1_Init 0 */
    CAN_FilterTypeDef sFilterConfig;
/* USER CODE END CAN1_Init 0 */
```



main.c(3)

```
/* USER CODE BEGIN CAN1_Init 2 */
/**-2- Configure the CAN Filter #####*/
sFilterConfig.FilterBank = 0;
sFilterConfig.FilterMode = CAN_FILTERMODE_IDMASK;
sFilterConfig.FilterScale = CAN_FILTERSCALE_32BIT;
sFilterConfig.FilterIdHigh = 0x0000;
sFilterConfig.FilterIdLow = 0x0000;
sFilterConfig.FilterMaskIdHigh = 0x0000;
sFilterConfig.FilterMaskIdLow = 0x0000;
sFilterConfig.FilterFIFOAssignment = CAN_RX_FIFO0;
sFilterConfig.FilterActivation = ENABLE;
sFilterConfig.SlaveStartFilterBank = 14;

if (HAL_CAN_ConfigFilter(&hcan1, &sFilterConfig) != HAL_OK)
{
    /* Filter configuration Error */
    Error_Handler();
}
```

main.c(4)

```
/*##-3- Start the CAN peripheral #####*/
if (HAL_CAN_Start(&hcan1) != HAL_OK)
{
    /* Start Error */
    Error_Handler();
}
```

```
/*##-4- Activate CAN RX notification #####*/
if (HAL_CAN_ActivateNotification(&hcan1, CAN_IT_RX_FIFO0_MSG_PENDING) != HAL_OK)
{
    /* Notification Error */
    Error_Handler();
}
```

main.c(5)

```
/*##-5- Configure Transmission process #####*/
TxHeader.StdId = 0x001;
//TxHeader.ExtId = 0x01;
TxHeader.RTR = CAN_RTR_DATA;
TxHeader.IDE = CAN_ID_STD;
TxHeader.DLC = 8;
TxHeader.TransmitGlobalTime = DISABLE;
/* USER CODE END CAN1_Init 2 */
```

main.c(6)

```
/* USER CODE BEGIN 4 */
void HAL_CAN_RxFifo0MsgPendingCallback(CAN_HandleTypeDef *hcan)
{
    /* Get RX message */
    if (HAL_CAN_GetRxMessage(hcan, CAN_RX_FIFO0, &RxHeader, RxData) != HAL_OK)
    {
        /* Reception Error */
        Error_Handler();
    }
    TxData[0] = RxData[0];
    TxData[1] = RxData[1];
    if (HAL_CAN_AddTxMessage(&hcan1, &TxHeader, TxData, &TxMailbox) != HAL_OK)
    {
        /* Transmission request Error */
        Error_Handler();
    }
}
/* USER CODE END 4 */
```

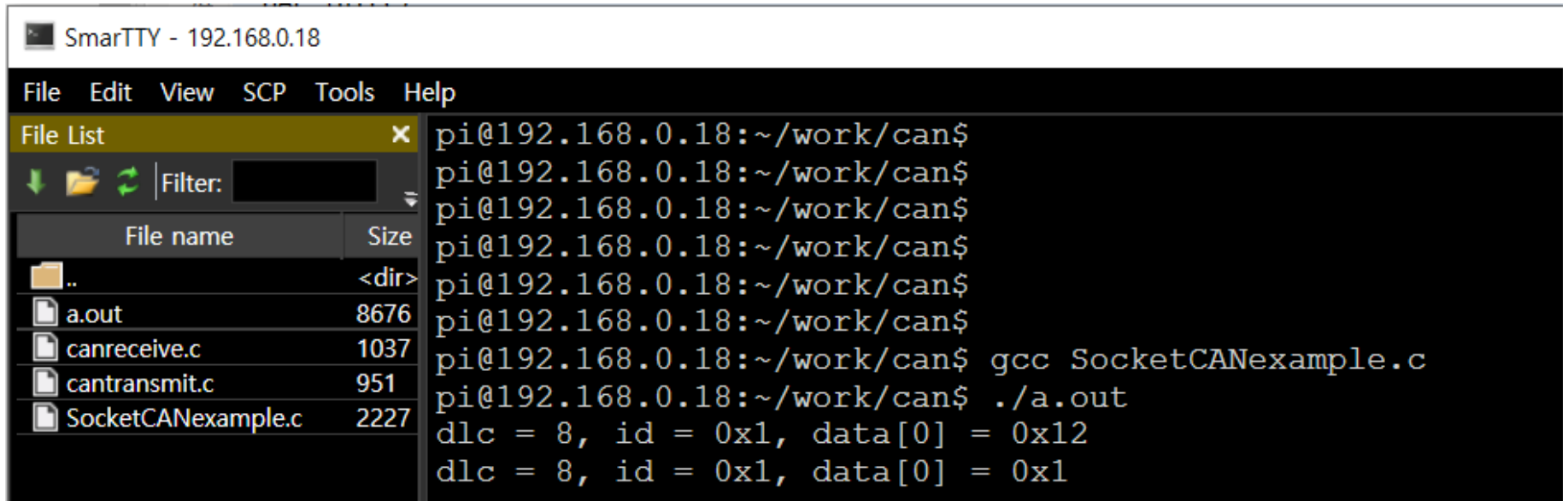
- Boot the Raspberry PI and run SocketCANexample.c

```
SmArTTY - 192.168.0.18
```

File name	Size
..	<dir>
a.out	8676
canreceive.c	1037
cantransmit.c	951
SocketCANexample.c	2227

```
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$
pi@192.168.0.18:~/work/can$ gcc SocketCANexample.c
pi@192.168.0.18:~/work/can$ ./a.out
```

▪ Run STM32 CAN example



The screenshot shows a terminal window with a menu bar (File, Edit, View, SCP, Tools, Help) and a file list on the left. The file list contains the following entries:

File name	Size
..	<dir>
a.out	8676
canreceive.c	1037
cantransmit.c	951
SocketCANexample.c	2227

The terminal output on the right shows the following commands and results:

```
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$  
pi@192.168.0.18:~/work/can$ gcc SocketCANexample.c  
pi@192.168.0.18:~/work/can$ ./a.out  
dlc = 8, id = 0x1, data[0] = 0x12  
dlc = 8, id = 0x1, data[0] = 0x1
```