USB Driver

MARVELL®

Driver Overview

- USB Driver Split into PC side and Device Side
- Device side matches PC side
- Current PC side driver is provided by SamSung
- Current Device side is developed by Marvell to cooperate with PC side driver.

Drivers supported by Marvell MIFI

PID	Description	Driver type composition	Use case	Group Comparison (reason)	remark
0x6861	integrated driver(RNDIS(#0,1) + UMS(#2)	RNDIS+UMS+ dynamic composition	MS Composite		Samsung integrated driver
0x6864	integrated driver(RNDIS(#0,1))	RNDIS+dynamic composition	MS Composite		

PID	Description	Driver type composition	Use case	Group Comparison (reason)	remark
N/A	RNDIS Only	RNDIS			
N/A	ECM Only	ECM			
N/A	UMS Only	UMS			

MARVELL

Dynamic composition

- Dynamic composition use the CDC-ACM Device Descriptor
- Enumerated as Modem in PC side
- 3 endpoints for one device: 1 for control, 2 for data(RX/TX)
- 2 Modems realized in current design: one for AT, one for Diag

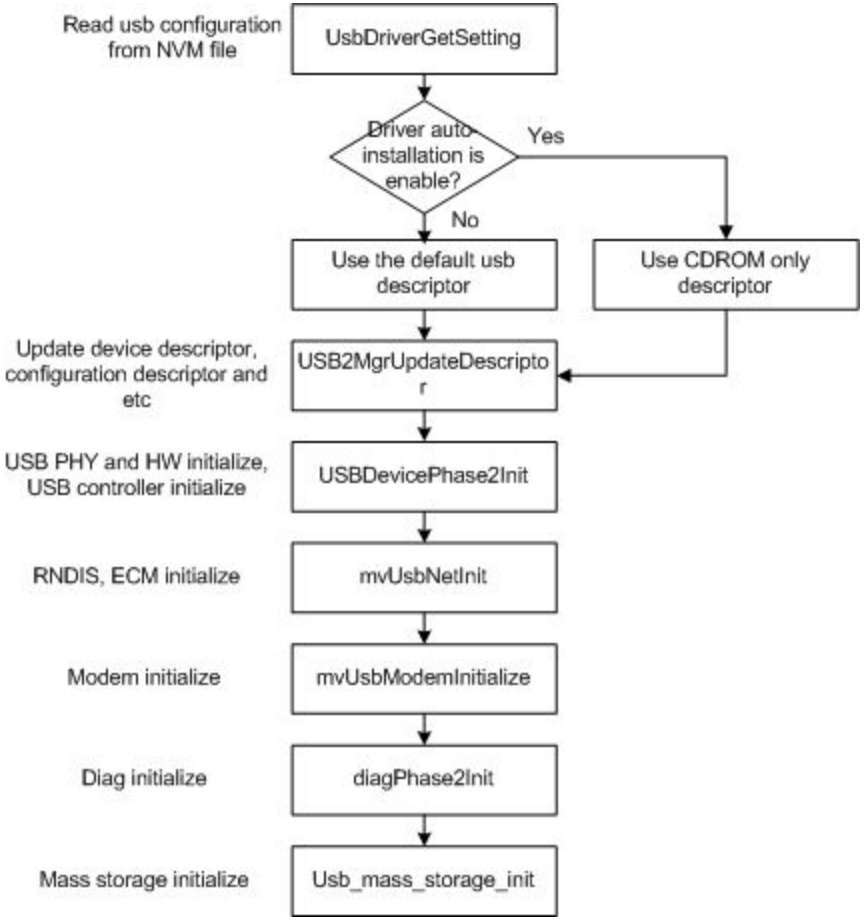
RNDIS

- Developed according to Microsoft RNDIS protocol
- 3 endpoints for one device: 1 for control, 2 for data(RX/TX)

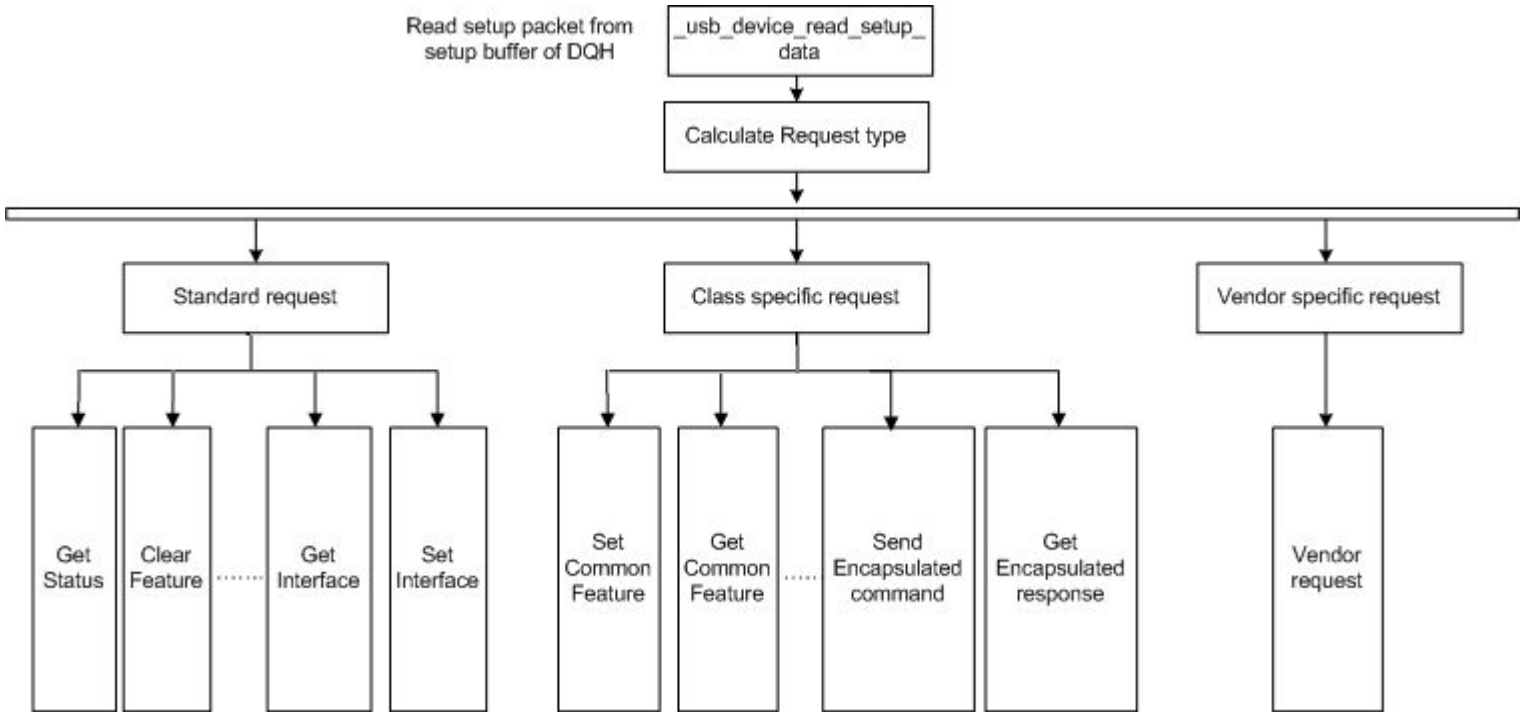
ECM

- Developed according to USB CDC-ECM protocol
- 3 endpoints for one device: 1 for control, 2 for data(RX/TX)

USB Initialization process



Setup packet process



How to update USB descriptor

- **Modify or add new descriptor in USB2MgrUpdateDescriptor.**
  - Device descriptor.
  - Configuration descriptor.
  - String descriptor.

```
void USB2MgrUpdateDescriptor(PlatformUsbDescType desc)
{
    UINT16 dev_desc_length,config_desc_length, qualif_desc_length, other_speed_desc_length;
    UINT16 devDesc_num, configDesc_num;
    BOOL storage = FALSE;

    ..................................

    /* Get Mass storage enable flag. */
    storage = usbCfg.mass_storage;

    switch(desc)
    {
        ......................

        case USB_SSG_MIFI_DESCRIPTOR: //Samsung MIFI
        {
            devDesc_num=0;
            //Device Descriptor
            devDesc[devDesc_num++] = 0x12; //  bLength             - Descriptor length
            devDesc[devDesc_num++] = 0x01; //  bDescriptorType     - Descriptor  Type
            ..................................

            configDesc_num=0;
            //Configuration Descriptor
            configDesc[configDesc_num++]   = 0x09; // bLength              - Descriptor length
            configDesc[configDesc_num++]   = 0x02; // bDescriptorType      - Descriptor Type
            ..................................

            break;
        }

        ......................

    } ? end switch desc ?

    ......................

    switch(desc)
    {

        ......................

        case USB_SSG_MIFI_DESCRIPTOR:
        case USB_AZW_MIFI_DESCRIPTOR:
        {
            USBDeviceSetDescriptor( USB_DESCRIPTOR_TYPE_STRING , strDescManufacturer, sizeof(strDescManufacturer), 1);//iManu
            USBDeviceSetDescriptor( USB_DESCRIPTOR_TYPE_STRING , strMobileDevice, sizeof(strMobileDevice), 2);//iProduct
            USBDeviceSetDescriptor( USB_DESCRIPTOR_TYPE_STRING , strMobileNumber, sizeof(strMobileNumber), 3);//iSerialNumber
            ......................
```

Device descriptor

Configuration descriptor

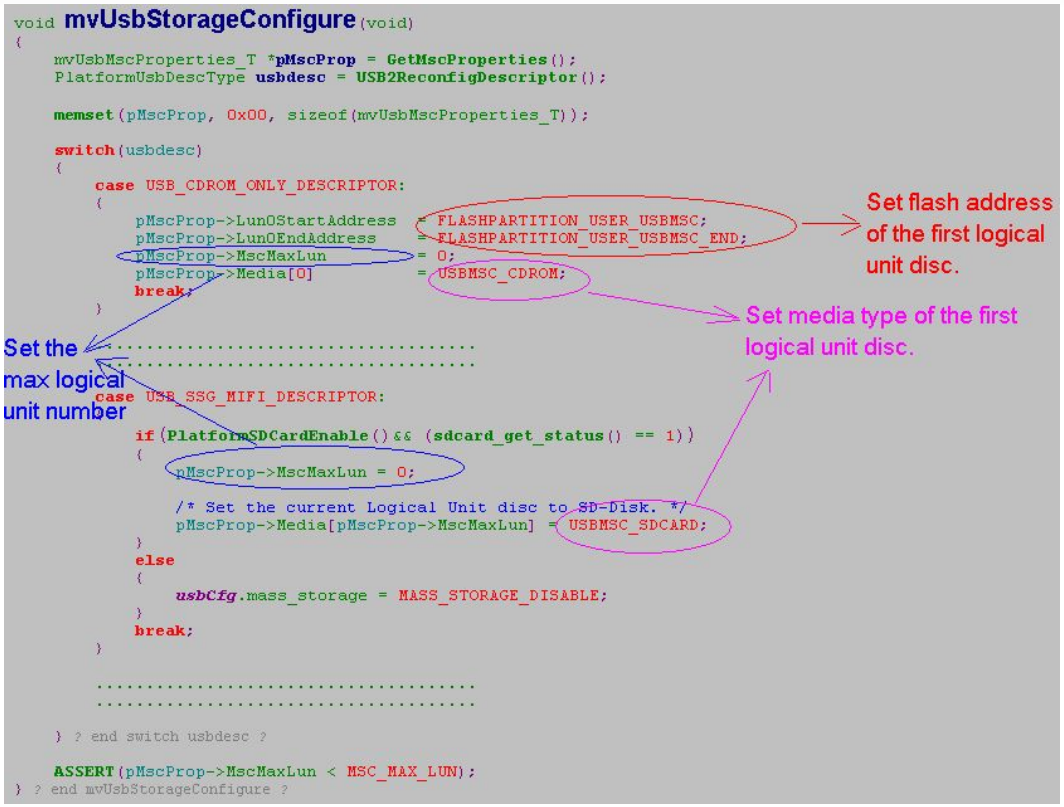
Set string descriptor

MARVELL®

How to configure Mass Storage

Configure mass storage.

- The max logical unit number.
- The Start/End flash address.
- The media type of logical disc.
- Enable/Disable Mass storage.




```
void mvUsbStorageConfigure(void)
{
    mvUsbMscProperties_T *pMscProp = GetMscProperties();
    PlatformUsbDescType usbdesc = USB2ReconfigDescriptor();

    memset(pMscProp, 0x00, sizeof(mvUsbMscProperties_T));

    switch(usbdesc)
    {
        case USB_CDROM_ONLY_DESCRIPTOR:
        {
            pMscProp->Lun0StartAddress  = FLASHPARTITION_USER_USBMSC;
            pMscProp->Lun0EndAddress    = FLASHPARTITION_USER_USBMSC_END;
            pMscProp->MscMaxLun         = 0;
            pMscProp->Media[0]          = USBMSC_CDROM;
            break;
        }
        ..................................
        ..................................
        case USB_SSG_MIFI_DESCRIPTOR:
            if(PlatformSDCardEnable() && (sdcard_get_status() == 1))
            {
                pMscProp->MscMaxLun = 0;

                /* Set the current Logical Unit disc to SD-Disk. */
                pMscProp->Media[pMscProp->MscMaxLun] = USBMSC_SDCARD;
            }
            else
            {
                usbCfg.mass_storage = MASS_STORAGE_DISABLE;
            }
            break;
        }

        ..................................
        ..................................

    } ? end switch usbdesc ?

    ASSERT(pMscProp->MscMaxLun < MSC_MAX_LUN);
} ? end mvUsbStorageConfigure ?
```

How to modify the flash address of mass storage

- **Modify the flash address of every logical unit disc in flashpartition.h**

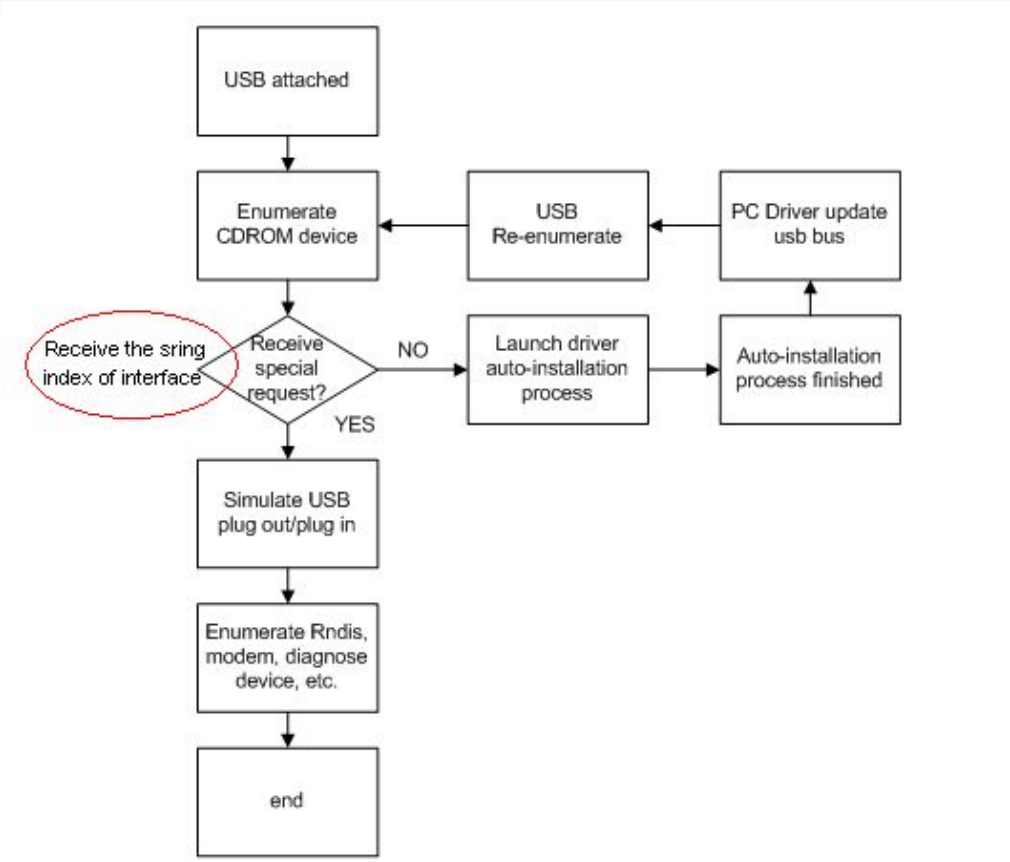
```
/*********************************************/
/* FAT System/Mass Storage Nor flash Address */
/*********************************************/

#ifdef SPI_NOR_FLASH

#define FAT_SYS_MAP_TABLE_ADDRESS           0x960000
#define FAT_SYS_START_ADDRESS               (0x960000+0x30000)
#define FAT_SYS_END_ADDRESS                 (FAT_SYS_START_ADDRESS+0x210000-1)

#define FLASHPARTITION_USER_USBMSC          0xBA0000
#define FLASHPARTITION_USER_USBMSC_END      (0xDA0000-1)

/* USB Flash Disc one*/
#define FLASHPARTITION_USER_USBMSC1         (0xDA0000)
#define FLASHPARTITION_USER_USBMSC1_END     (0xEA0000-1)

/* USB Flash Disc two*/
#define FLASHPARTITION_USER_USBMSC2         (0xEA0000)
#define FLASHPARTITION_USER_USBMSC2_END     (0xFA0000-1)

#define FLASHPARTITION_PSM_ADDR             0xFA0000

#else /* Nand Flash layout*/
```

the first logical unit disc

the second logical unit disc

the third logical unit disc

Driver Auto-installation

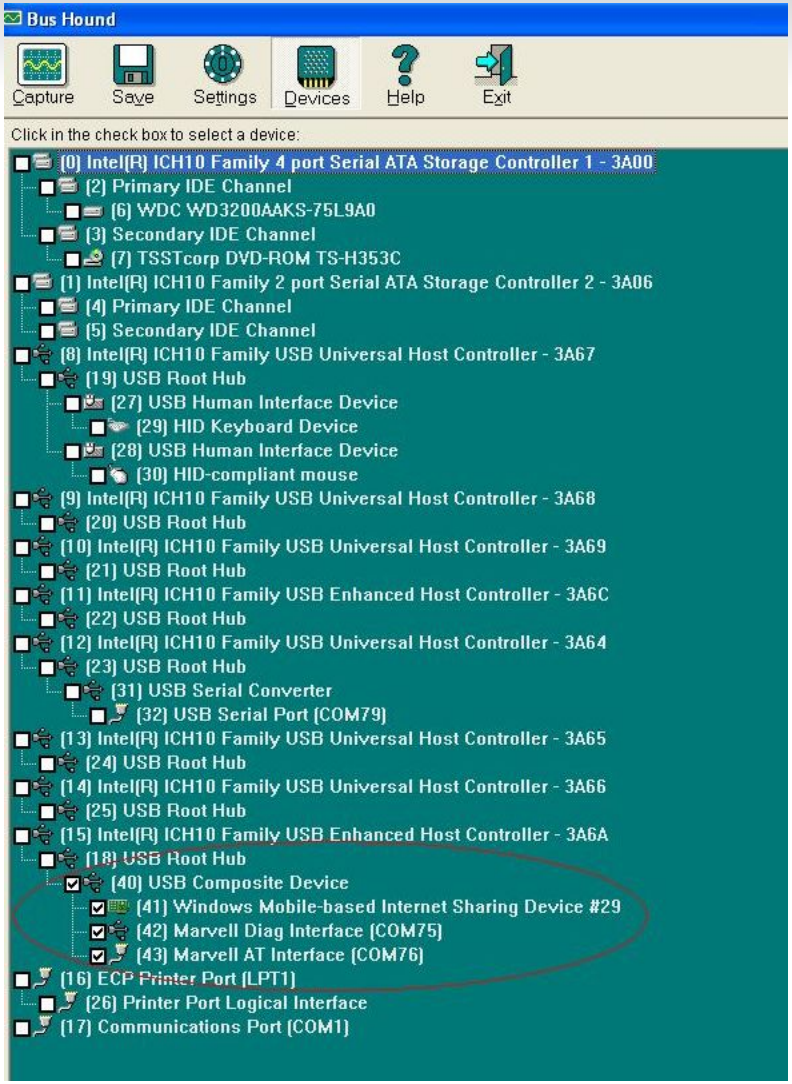


Driver auto-installation flow chart

Debug With Bushound

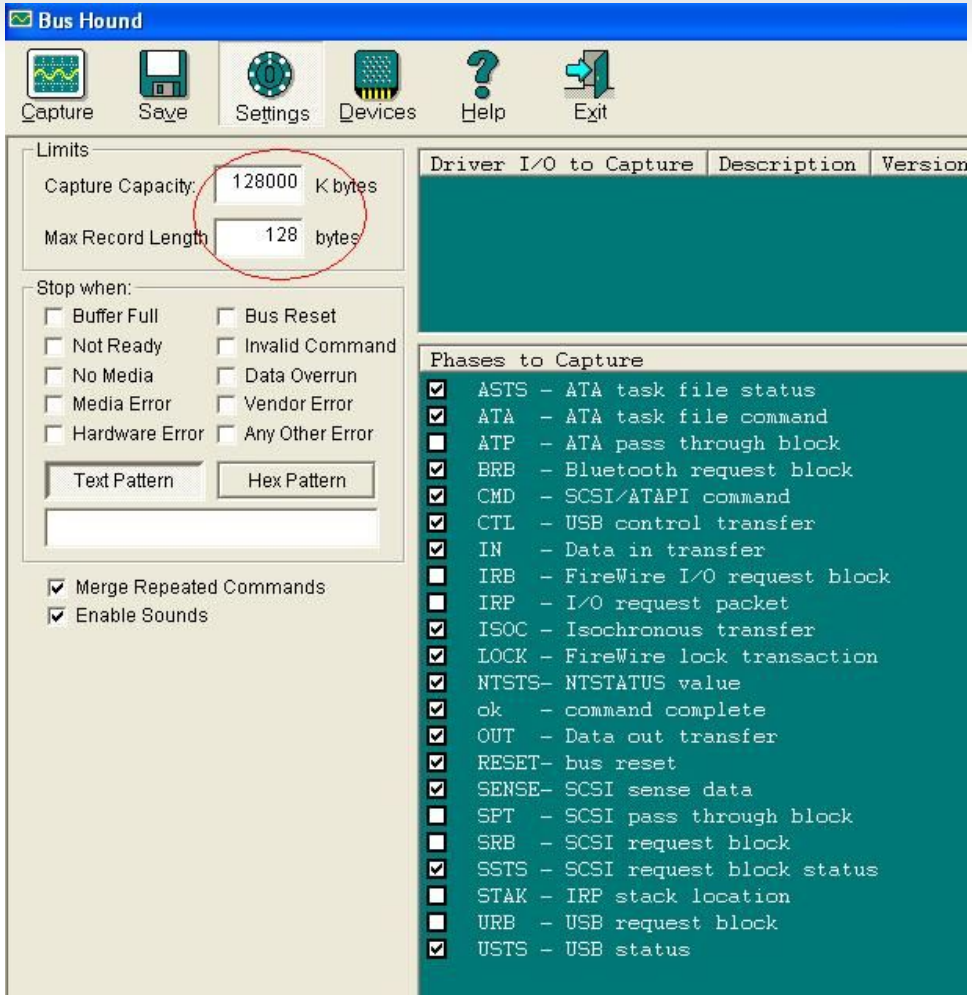
How to capture bushound log

- Select usb devices



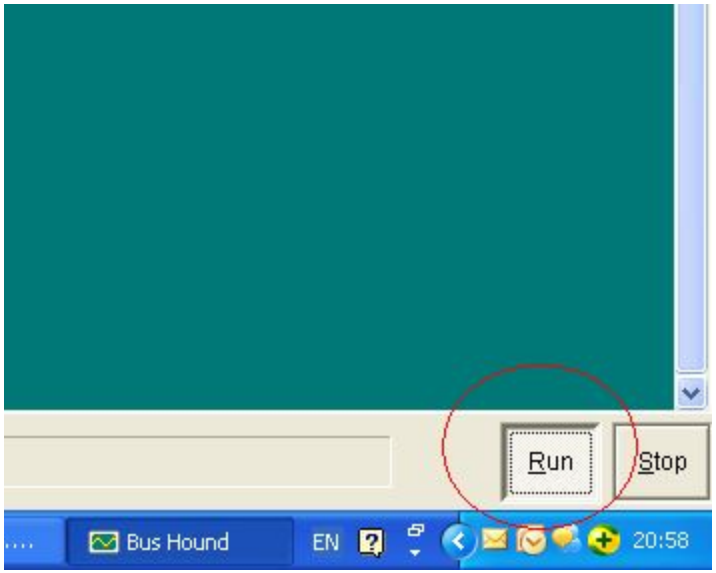
How to capture bushound log

- Set “Capture Capacity”.
- Set “Max Record Length”



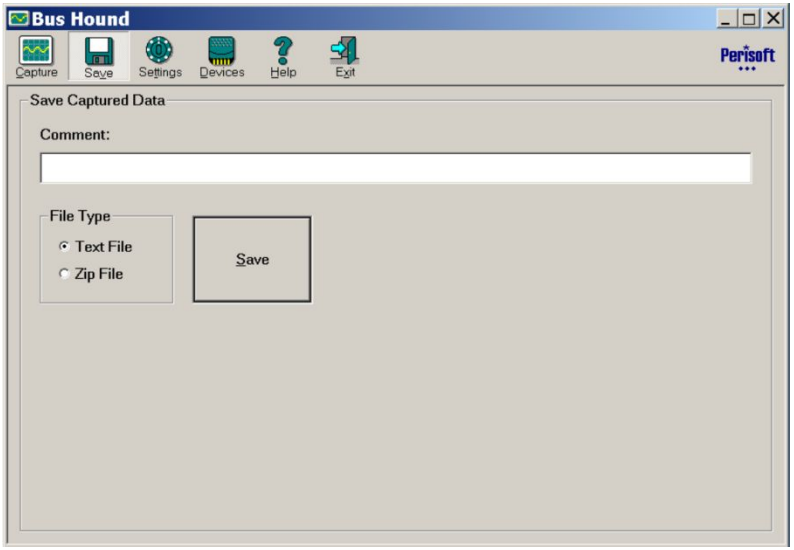
How to capture bushound log

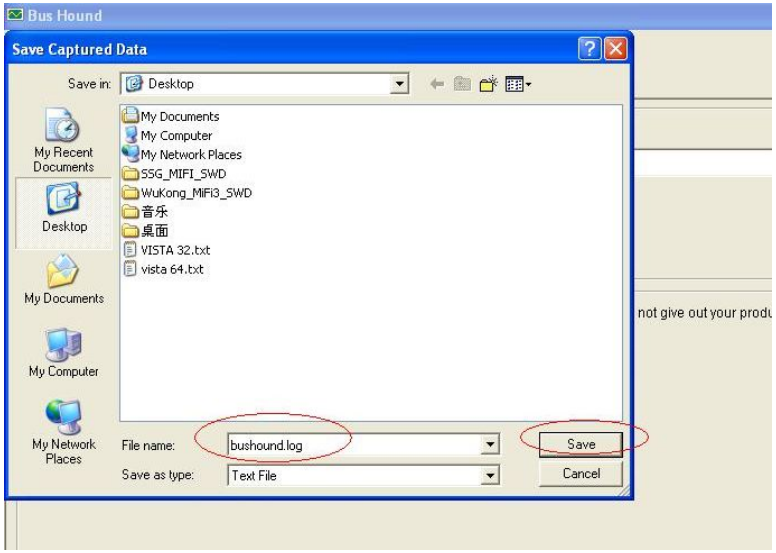
- Click “run” button to start to capture log.



How to capture bushound log

Save log





Thank You!