#11897 [HD] XJMF communication between TP and UW/RW made by Company Ho

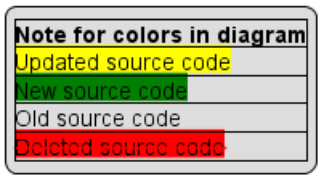
Change log

Rev.	Date	Author	Details
1	2023/01/10	GCS	Created

Target System

TP-J520HD V3.??JXD0xx

Note for diagrams



100. Plug-in for sending and receiving XJMF

1. Description

100-01.

The XJMF sending/receiving plug-in (UnwinderManager plugin) that was implemented in #7041 is ported to HD, and expanded/changed to the specifications described later.
Change the plugin name to “UwRwManager plugin”

100-02.

XJMF sending/receiving plug-in performs HTTP communication with UW/RW in XJMF format and acquires the following information.

	UW	RW
Status	○	○
Remaining amount of paper	○	×
Windable amount of paper	×	○
Currently roll diameter	○	○
Stop roll diameter	○	○
Media name	○	○
Paper width	○	○
Paper weight	○	○
Paper thickness	○	○
UW dancer web tension	○	×
RW dancer web tension	×	○
RW winding tension	×	○

100-05.

The controller and UW/RW port numbers and IP addresses are defined in the TP-UW_Communication.ini file.
※Please refer to【TP-UW_Communication.ini】(See below).

- In #7041, the file name was “TP-UW_Communication.ini”, but the file name was changed.
- But refer to the existing key below for the IP address of the controller.
- File path: D:\TPJ_xx\Client_xx\Preferences\PrinterDescriptor.ini
- Section: CLIENT_PC_SELF
- Key: IP_ADDR

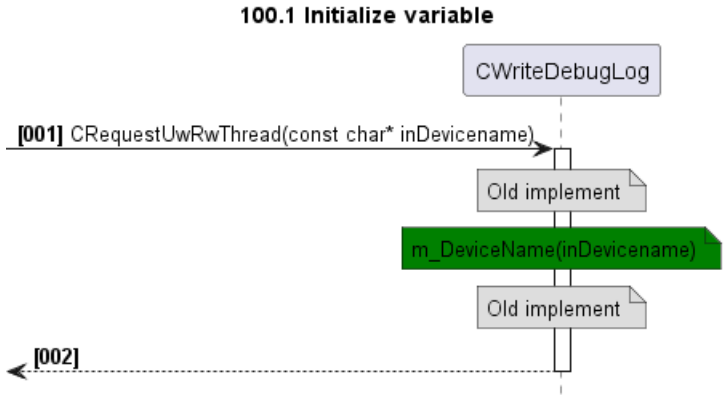
2. Solution

- Rename plugin UnwinderManager to UwRwManager.
- In plugin UwRwManager, replace “Uw” in classes name by “UwRw”.
- In plugin UwRwManager, replace “Uw” in methods name by “UwRw”.
- in class CRequestUwRwThread, add variable to store name of UW or RW.

dataIF.h

```
// before
UINT m_PaperReceivingTimeId;
UINT m_StatusReceivingTimeId;

// after
UINT m_PaperReceivingTimeId;
UINT m_StatusReceivingTimeId;
const char* m_DeviceName;
```

3. Detail implementation



103. Xml output of send and receive contents

1. Description

103-01.

Place the xml output of the sent / received contents directly under the folder in the following path.

- D:\TPJ_xx\Client_xx\Log\UW_CONNECT\yyyymmdd.xml
- D:\TPJ_xx\Client_xx\Log\RW_CONNECT\yyyymmdd.xml

103-02.

The output xml file is automatically deleted after a certain period of time.

- UWandRW_Receiver.exe checks the above storage period at startup and deletes it if necessary.
- The retention period of the xml file is defined in the “TP-UWRW_Communication.ini” file.

2. Solution

- In class CWriteDebugLog, add variable to store log folder path for RW.

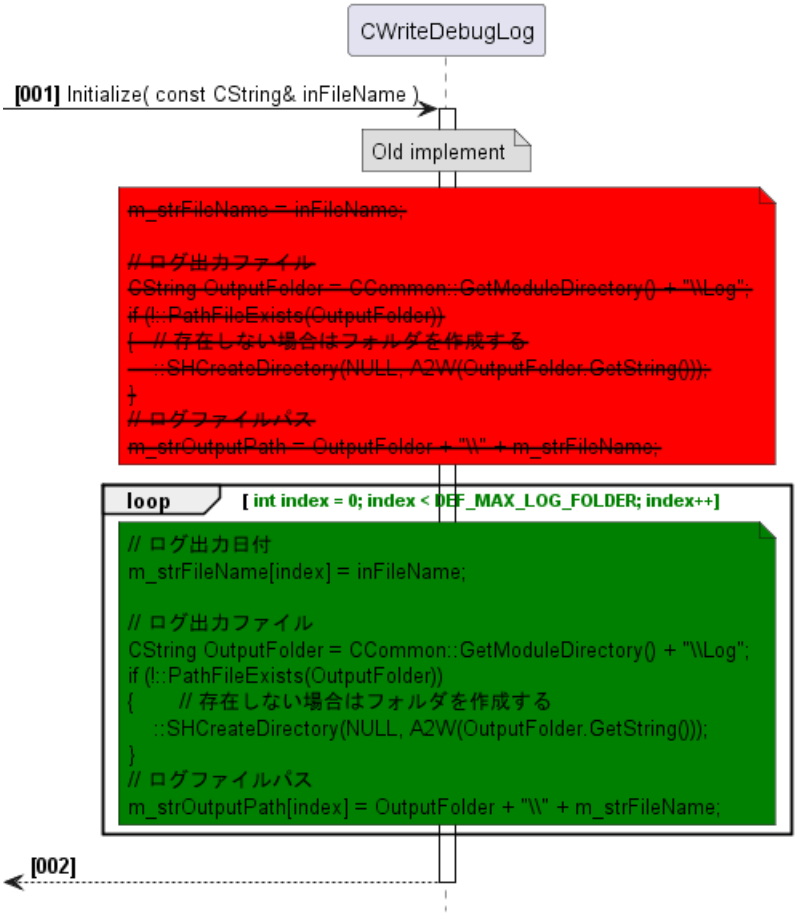
UWandRW_Receiver\WriteDebugLog.h

```
// before
CString     m_strFileName;      //!< ログ出力フォルダの日付
CString     m_strDate;          //!< ログ出力フォルダの日付
CString     m_strOutputPath;    //!< ログ出力ファイルパス

// after
CString     m_strFileName[DEF_MAX_LOG_FOLDER];      //!< ログ出力フォルダの日付
CString     m_strDate[DEF_MAX_LOG_FOLDER];          //!< ログ出力フォルダの日付
CString     m_strOutputPath[DEF_MAX_LOG_FOLDER];    //!< ログ出力ファイルパス
```

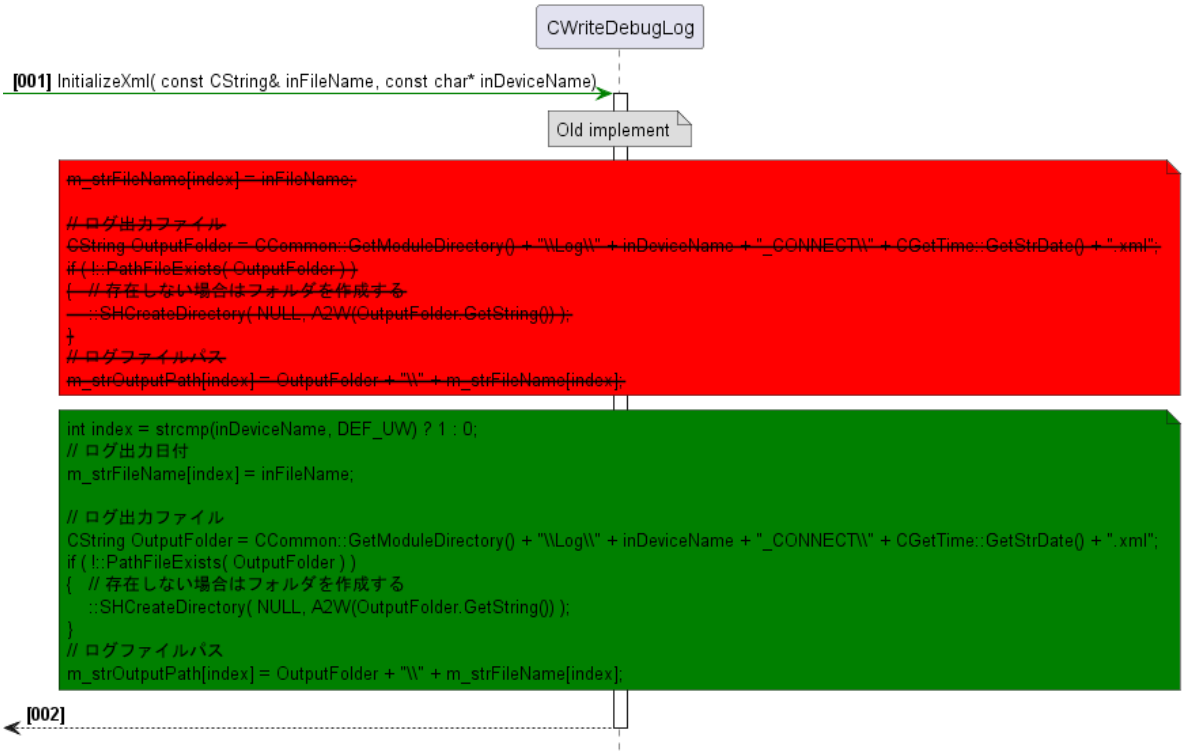
- In method CWriteDebugLog::InitializeXml, append set log folder path for RW.
- In method CXmlLogData::GetLogFolderPath, append create Log folder path for RW.
- In method CMainFunction::DeleteLogFolder, append delete log folder for RW.

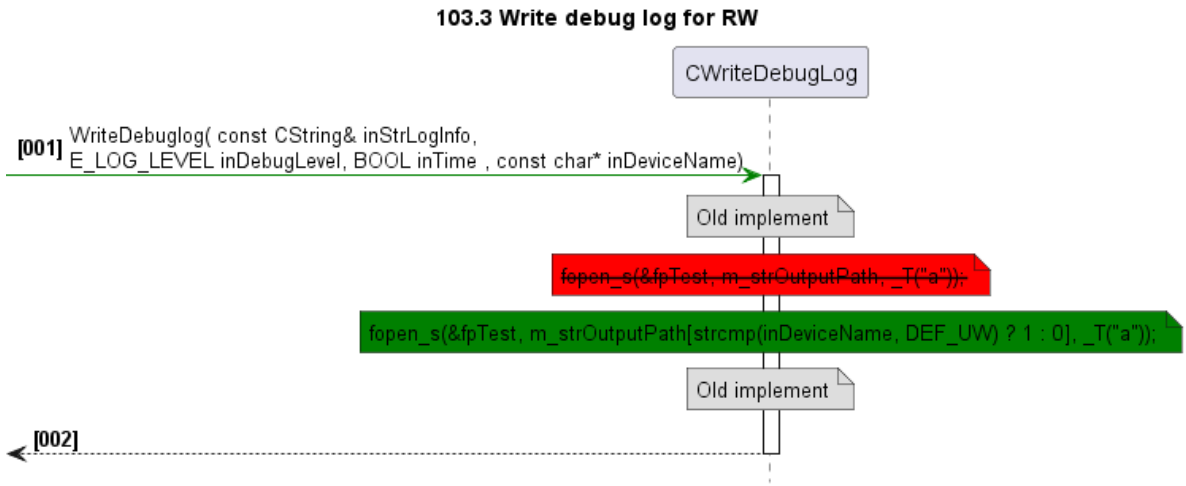
3. Detail implementation

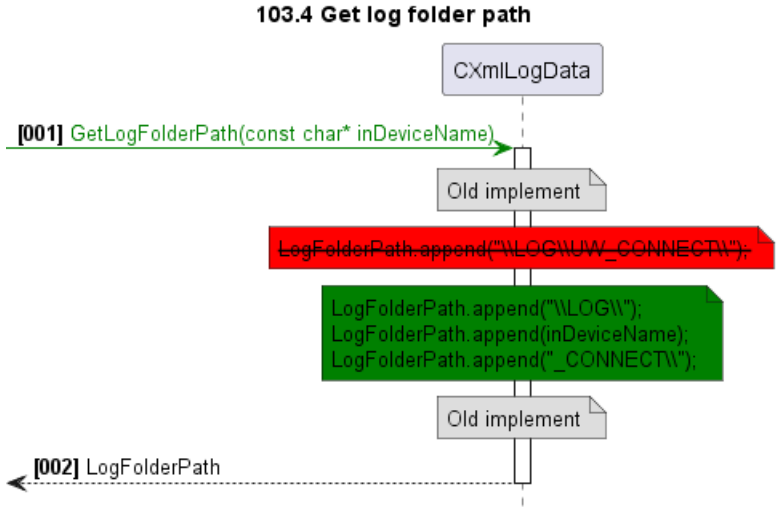
103.1 Initialize log folder path for RW

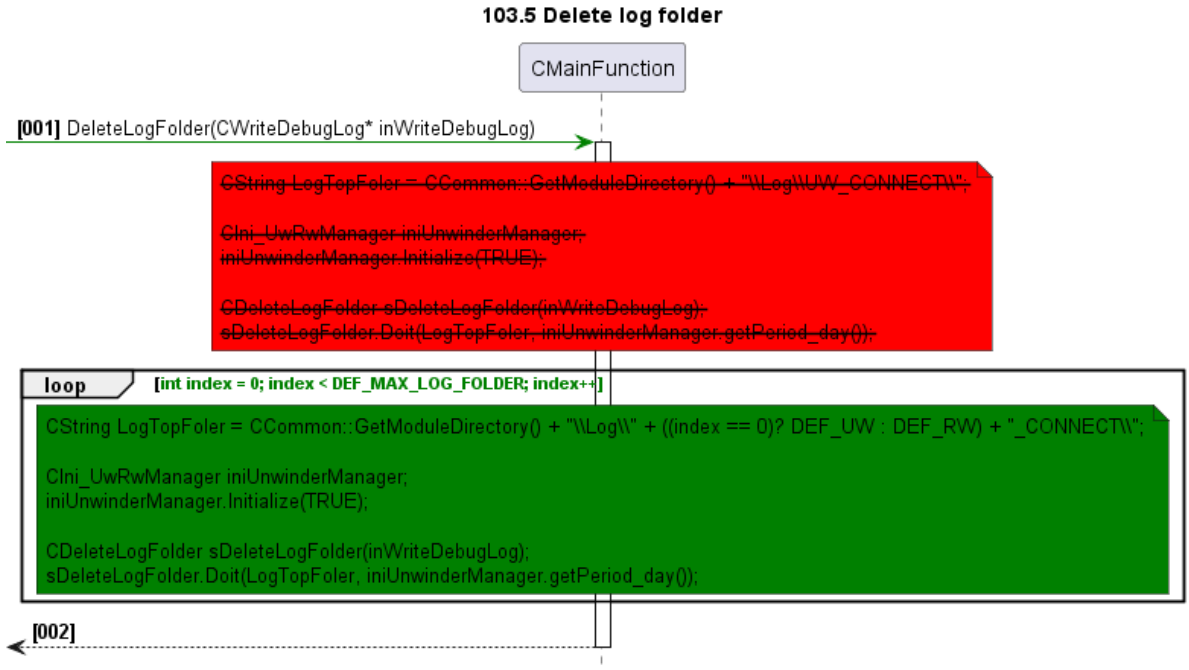


103.2 Set log folder path for RW









200. Enable/disable of XJMF communication with UW/RW

1. Description

200-01.

201-4xx XJMF communication specifications work only when the following key is set to 1.

- [File name]: PrinterDescriptor.ini
- [Section name]: OPTION
- [Key name: 1]: UW_CONNECT_FUNCTION
- [Key name: 2]: RW_CONNECT_FUNCTION
    - The default value for the above keys is 0.
    - Manually change to 1 during setup when communication with UW/RW made by Ho.
    - It is also possible to communicate with only one of UW and RW

2. Solution

- In class CIni_PrinterDescriptor, add method to get value of RW_CONNECT_FUNCTION key from PrinterDescriptor.ini file.

Common\Ini_PrinterDescriptor.h

```
public:
...
/**
    @brief      Get Rewinder Option
    @retval 1: enable RewinderManager function
    @retval 0: disable RewinderManager function
*/
UINT getRewinderOption();
```

Common\Ini_PrinterDescriptor.cpp

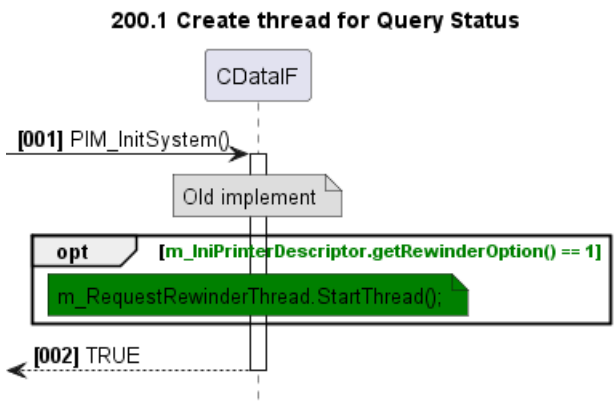
```
UINT CIni_PrinterDescriptor::getRewinderOption()
{
    return GetValueInt("OPTION", "RW_CONNECT_FUNCTION", 0);
}
```

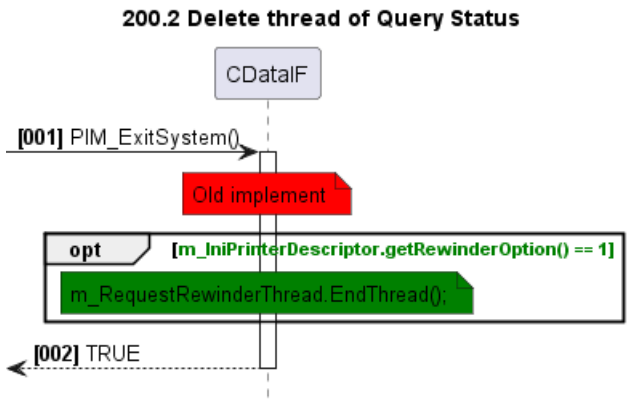
- In CDataIF class, add new variable to handle Rewinder thread.

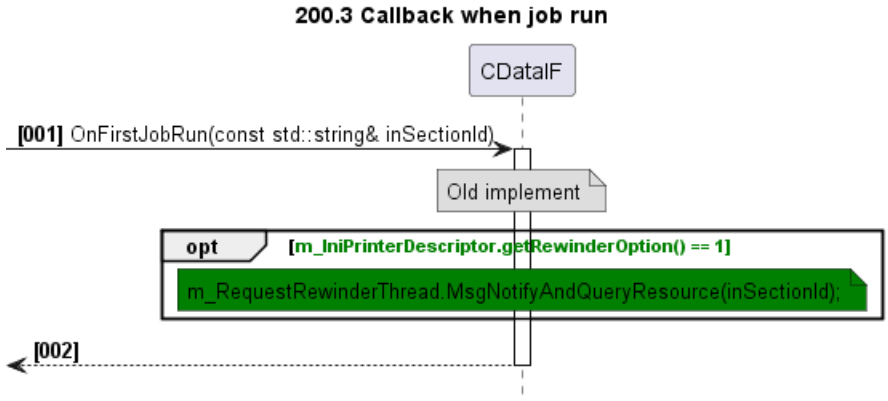
Common\Ini_PrinterDescriptor.cpp

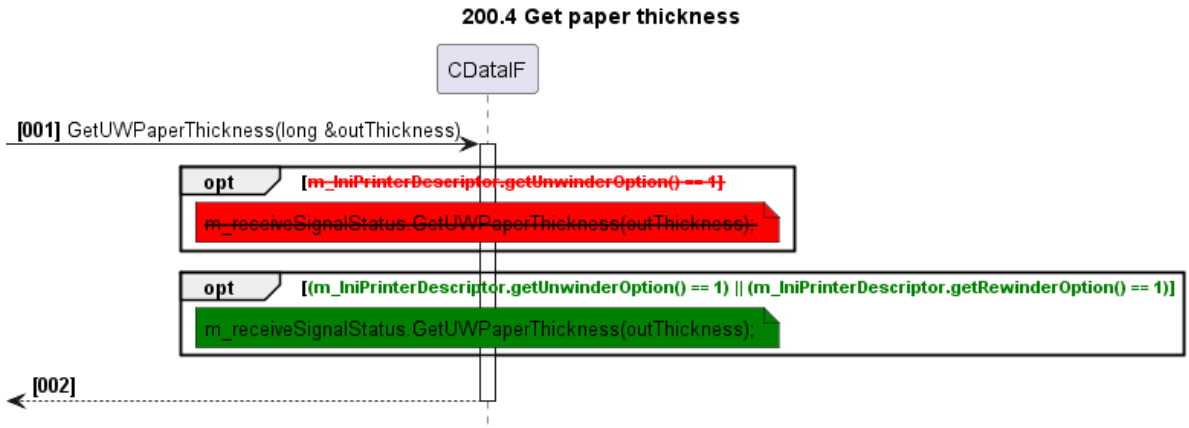
```
// before
unwinder_manager::CRequestUwRwThread m_RequestUnwinderThread;

// after
uwrw_manager::CRequestUwRwThread m_RequestUnwinderThread;
    uwrw_manager::CRequestUwRwThread m_RequestRewinderThread;
```

3. Detail implementation









300. Status monitoring channel (channel registration)

1. Description

300-01.

The controller registers the communication channel for status monitoring in order to obtain status information from UW/RW.

- Channel registration is done for each of UW and RW.

300-03.

Channel registration is performed by notifying the UW/RW of the “QueryStatus” command by specifying Subscription from the controller.

- For Subscription, specify the following information.

    - URL (information notification destination)
    - RepeatTime (Regular notification interval from UW/RW)

- The specified value is defined in “TP-UWRW_Communication.ini”.

300-04.

If channel registration is successfully completed (returncode=0), Receive channel ID (refID) in response from UW/RW.

- Store the received channel ID in “Status_CannelID” of “TP-UWRW_Communication_work.ini”

300-05.

If there is no channel registration result response from UW/RW, or if the channel registration result is abnormal (returncode=other than 0), it is determined that UW/RW is not activated.

- At this time, store “” (empty value) in “Status” of “TP-UWRW_Communication_work.ini”

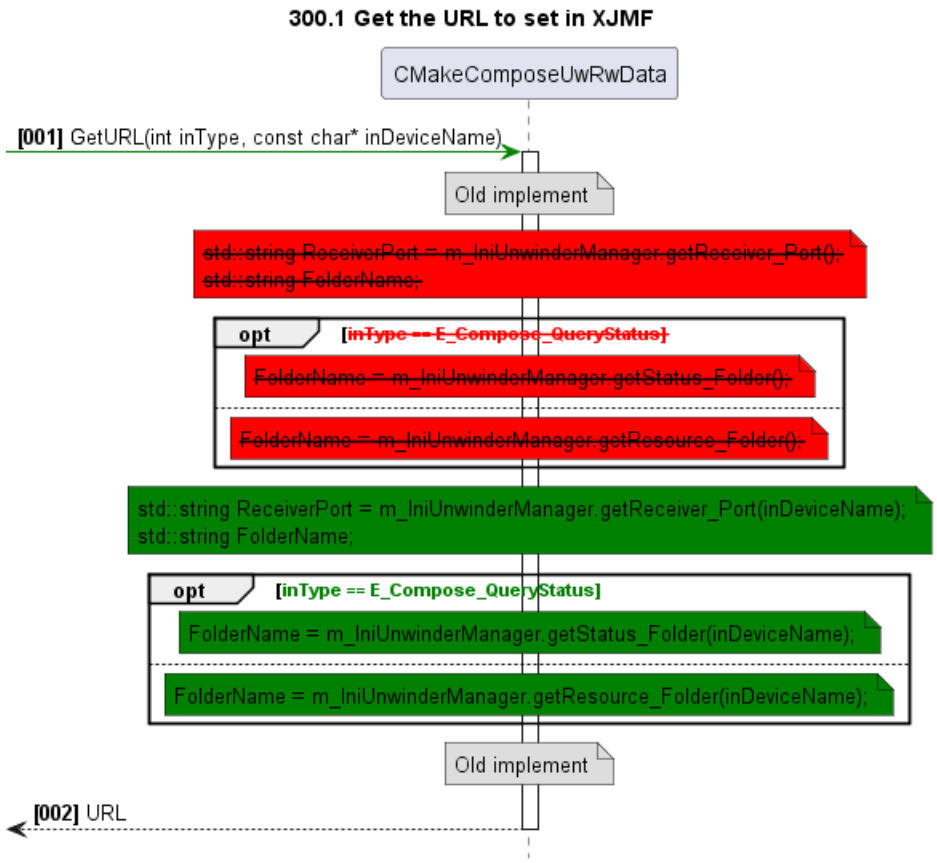
300-06.

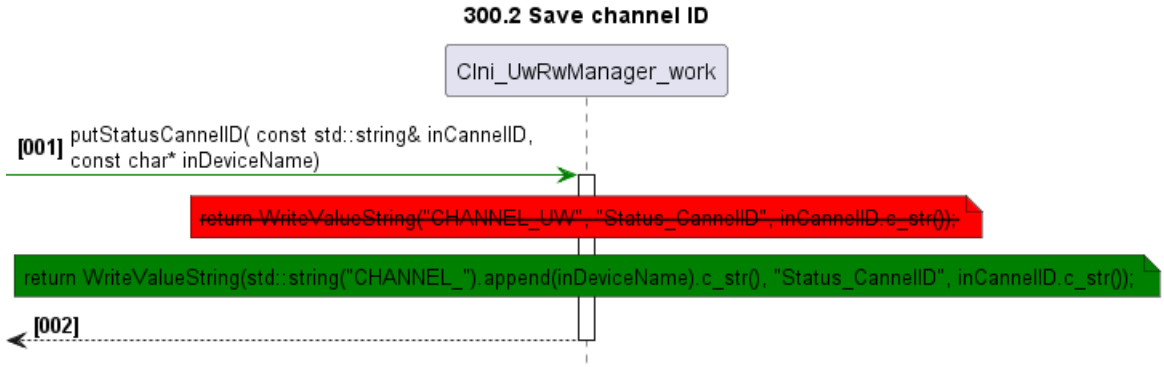
If there is no channel registration result response from UW/RW, a Ping confirmation timer is issued and a reconnection request is made.
If the ping passes, the channel registration for the status monitoring channel is performed until the channel registration is completed.

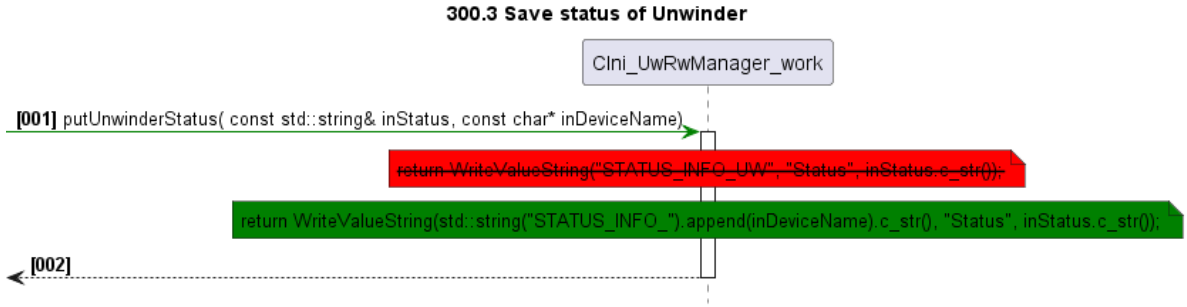
2. Solution

- Status monitoring channel:
    - In CRequestUnwinderThread::ThreadProc() method, call to RequestQueryStatus() to request status of RW.
- In CMakeComposeUwRwData::GetURL() method, append create URL for RW query status.
- In CIni_UwRwManager_work::putStatusCannelID() method, append saving channel ID response from RW.
- In CIni_UwRwManager_work::putUnwinderStatus() method, append saving Status for RW.

3. Detail implementation







301. Status monitoring channel (channel deletion)

1. Description

301-01.

Deletion of the registered channel is performed by notifying the UW/RW of the

“StopPresChParam” command from the controller.

- Specify the channel ID at the time of channel registration to “ChannelID” of the “StopPresChParam” command
- Obtain the channel ID when registering a channel from “Status_CannelID” in “TP-UWRW_Communication_work.ini”
- When receiving a successful channel deletion response, also delete “Status_CannelID” in “TP-UWRW_Communication_work.ini” (store a empty value).

301-02.

When registering a channel, if the channel information has already been registered, delete the channel first.

- Applicable when there is a value in “Status_CannelID” in “TP-UWRW_Communication_work.ini”

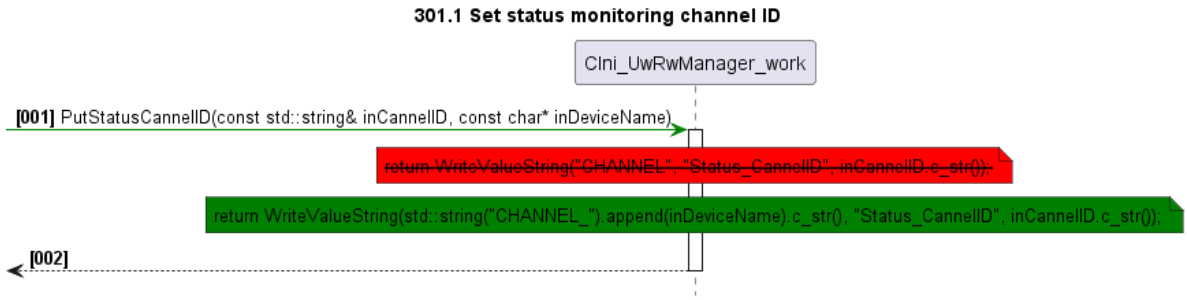
301-03.

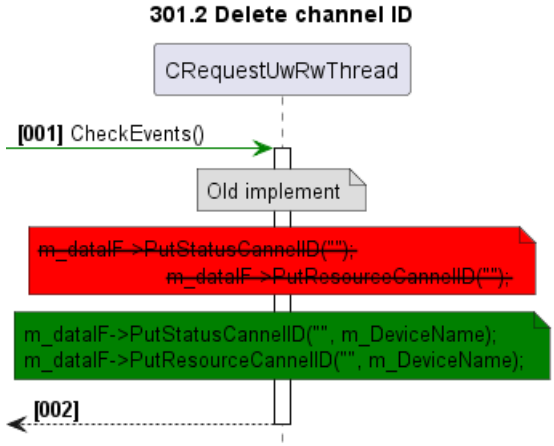
- When the controller ends, delete the channel first.

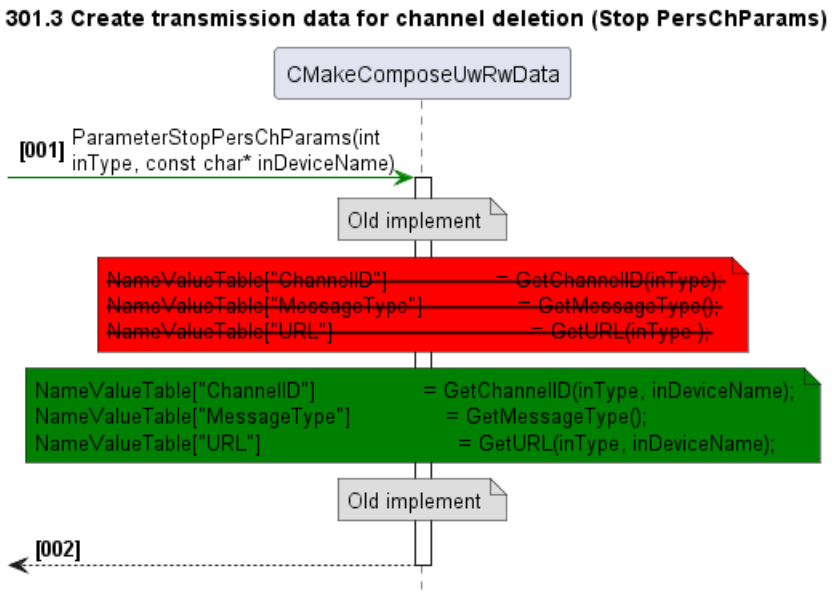
2. Solution

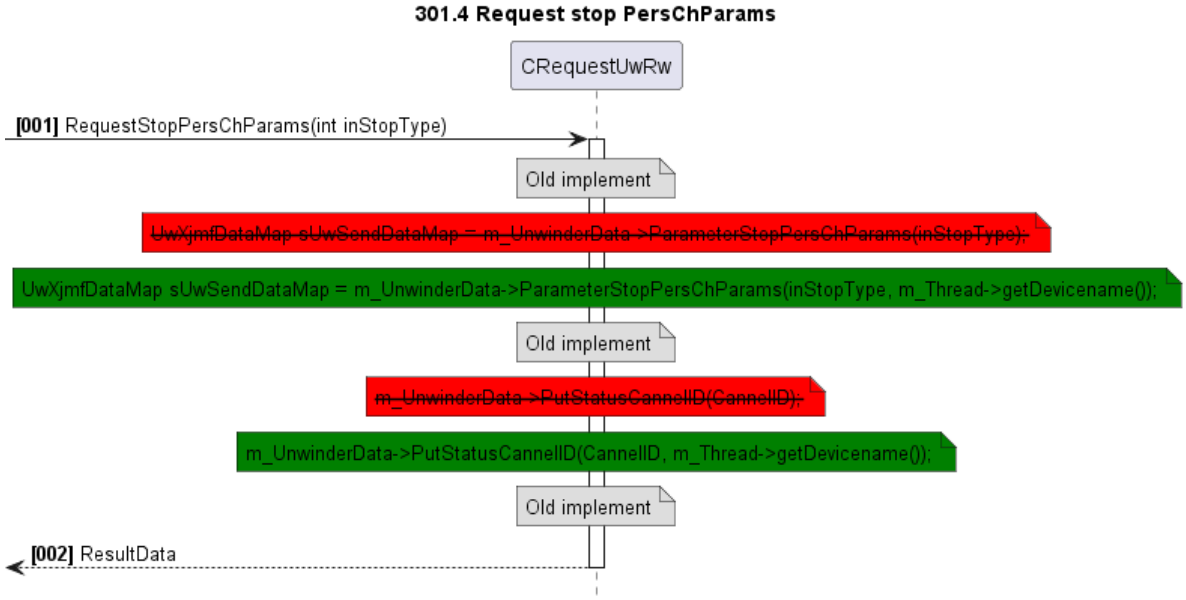
- In CIni_UwRwManager_work::putStatusCannelID method, append set Status_CannelID for RW.
- In CRequestUwRwThread::ThreadProc() method, check for existing channel IDs in TP-UW_Communication_work.ini and delete them.
- In CRequestUwRwThread::CheckEvents() method, when WM_TIMER message is received, delete the channel IDs in TP-UWRW_Communication_work.ini file.
- In CRequestUwRwThread::ThreadProc() method, wait for process “UWandRW_Receiver.exe” to end or m_ExitThread set.
- In CRequestUwRwThread::ThreadProc() method, call to RequestStopPersChParams() for channel which has been registered.

3. Detail implementation









302. Status monitoring channel (receiving periodic notifications)

1. Description

302-01.

When the channel registration is performed normally, the UW/RW notifies the “SignalStatus” command at the specified interval. The controller receives this.

302-02.

Receive the following information with the “SignalStatus” command from UW/RW and store the information

- DeviceInfo/Status (UW/RW device status)
  - Store the received value in “Status” of “TP-UWRW_Communication_work.ini”
  - UW/RW device status list

Status value	Description
Idle	Standby
NonProductive	Preparing for interlocking operation
Setup	Arm setup status, roll setup status
Offline	Operation preparation incomplete state
Production	In transit
Stopped	Dancer tension released

302-03.

If there is no “SignalStatus” from the UW/RW after the interval + α seconds specified when the controller registers the status monitoring channel, it determines that the UW/RW has not started.

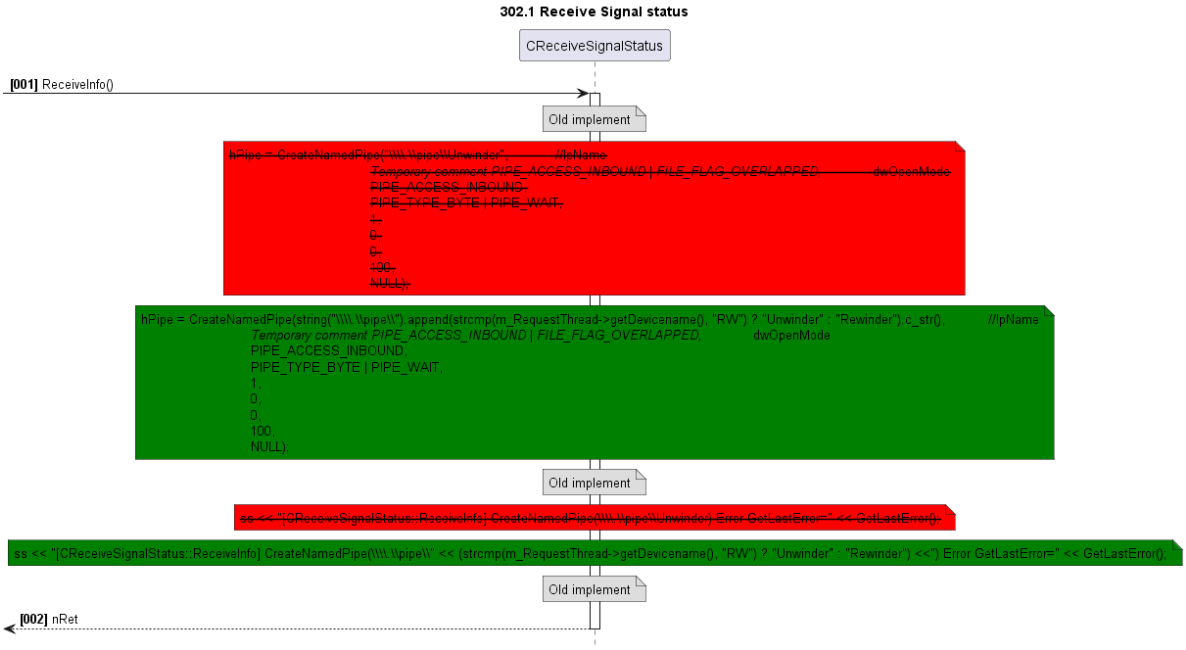
- +α seconds is defined in the “TP-UWRW_Communication.ini” file.
- At this time, store “” (empty value) in “Status” of “TP-UWRW_Communication_work.ini”
- At this time, the following warning message dialog is displayed.
  - For UW
    - (Ja) UWとの通信エラーが発生しました。
    - (En) A communication error with UW has occurred.
  - For RW
    - (Ja) RWとの通信エラーが発生しました。
    - (En) A communication error with RW has occurred.
- At this time, a Ping confirmation timer is issued and a reconnection request is issued.

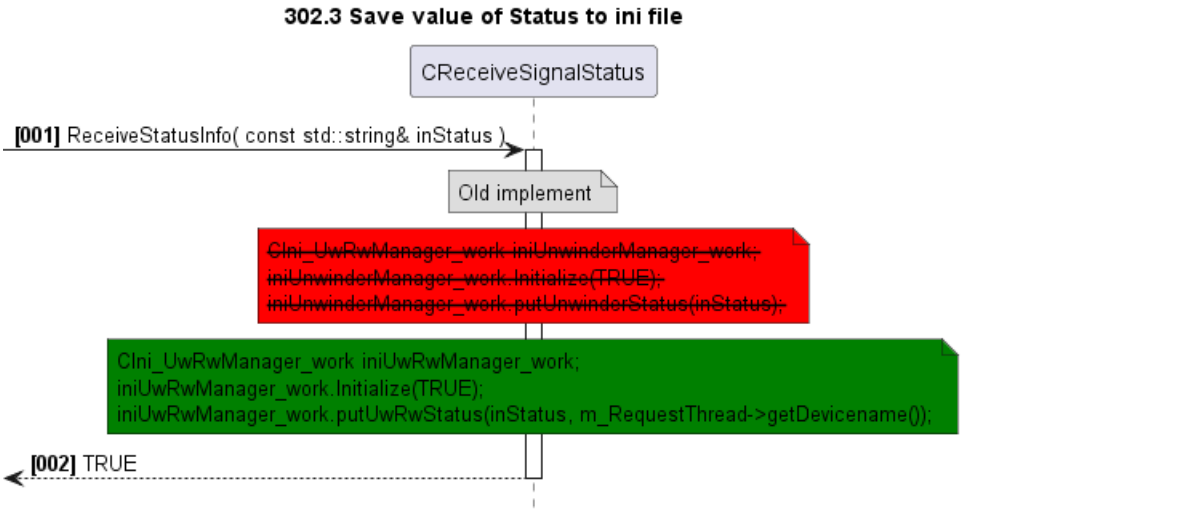
If the ping passes, the channel registration for the status monitoring channel is performed until the channel registration is completed.

2. Solution

- In method CDataIF::UpdateDisplayUWStatus(), post a message about RW status and display the warning dialog when RW is offline
  - In method CRequestUwRwThread::CheckReceiverRunning(), if UWandRW_Receiver.exe is not run, call method UpdateDisplayUWStatus(false) to display UW offline icon.
  - In method CRequestUwRwThread::CheckUWStatus(),
    - If receive the response from RW successfully, call method UpdateDisplayUWStatus(true) to display RW online icon.
    - Else, call method UpdateDisplayUWStatus(false) to display RW offline icon.
  - In method CReceiveSignalStatus::ReceiveStatusInfo:
    - Stop the current timeout timer and start a new one.
    - Set the RW status into UnwinderManager_work.ini file.
    - call method UpdateDisplayUWStatus(true) to display RW online icon.

3. Detail implementation





400. Paper information notification channel (channel registration)

1. **Description**

   400-02.

   The channel registration timing is set immediately after the controller notify printing condition information to UW/RW by “CommandResource” command and receives a response from the UW when the controller is started.

   However, the process of notifying “CommandResource” to UW/RW is will be addressed in the next “Order 3”.

   Therefore, at the time of “Order 2”, the channel registration timing shall be the time when the controller starts up.

   400-04.

   If channel registration is successfully completed (returncode=0 or returncode=18), receive channel ID (refID) in response from UW/RW.

   - Store the received channel ID in “Resource_CannelID” of “TP-

UWRW_Communication_work.ini”

400-05.

In the case of channel registration with UW, the presence or absence of the paper thickness gauge option is obtained from the return code upon completion. Not applicable for channel registration with RW.

- If returncode=0, determine that there is a paper thickness gauge option, and store 1 in “Enable_ThicknessMeasureOP” of “TP-UWRW_Communication_work.ini”
- If returncode=18, determine that there is no paper thickness gauge option and store 0 in “Enable_ThicknessMeasureOP” of “TP-UWRW_Communication_work.ini”

2. Solution

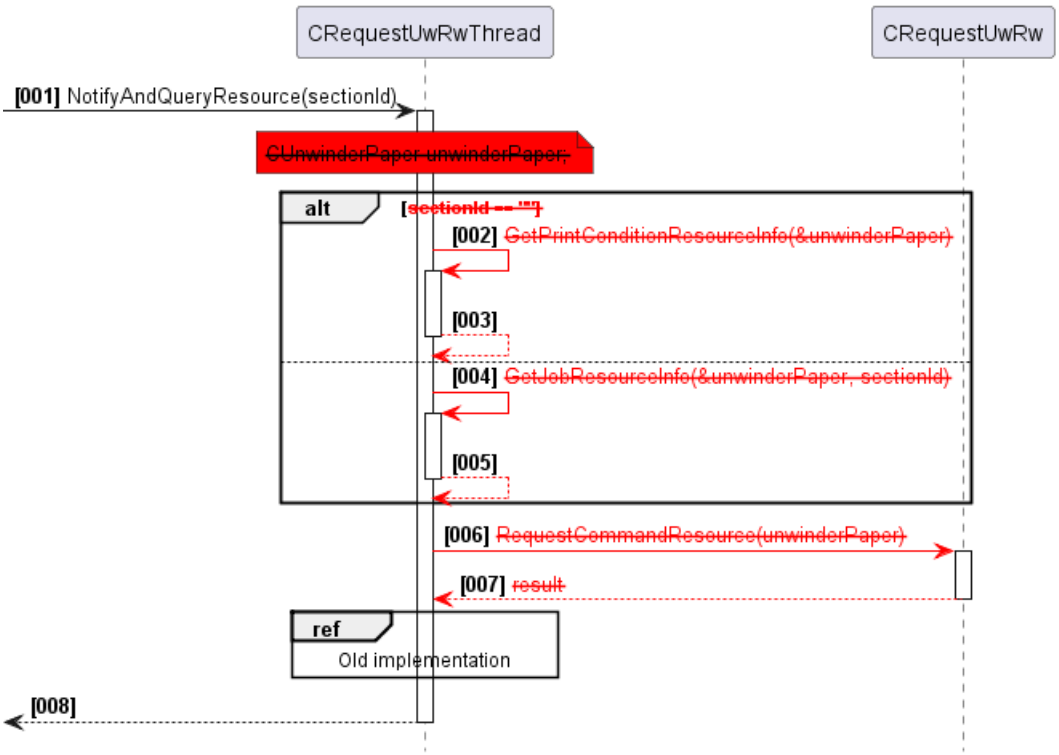
- In method CRequestUwRwThread::NotifyAndQueryResource(), temporarily comment out call to CRequestUwRw::RequestCommandResource().
- In plugin PrintConditionGUI and JobPrintSequence, temporarily comment out call to SUnwinderManager_Callbacks callback functions.
- In method CRequestUwRw::RequestQueryResource(), store the received channel ID when returncode=18.
- In method CRequestUwRw::RequestQueryResource(), store the received Enable_ThicknessMeasureOP value when returncode=0 or returncode=18.
- In method CIni_UwRwManager_work::putResourceCannelID(), add parameter inDeviceName.
- Add method CIni_UwRwManager_work::putEnable_ThicknessMeasureOP() to store the value to ini file.

Ini_UWRWManager_work.h

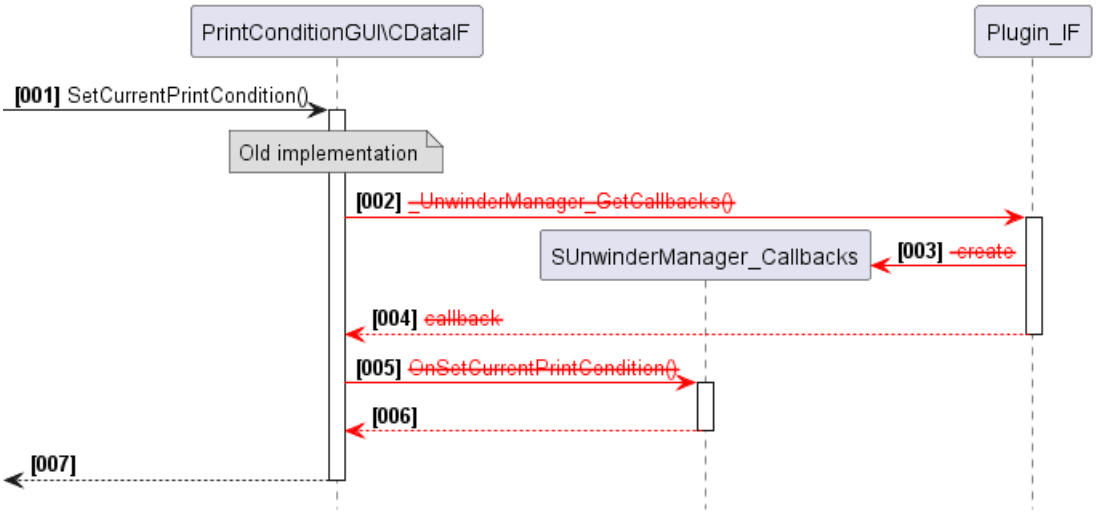
```
// before
class CIni_UwRwManager_work
{
    BOOL putResourceCannelID( const std::string& inCannelID);
}

// after
class CIni_UwRwManager_work
{
    BOOL putResourceCannelID( const std::string& inCannelID, const char* inDeviceName);
    ...
    BOOL putEnable_ThicknessMeasureOP( const std::string& inEnable_ThicknessMeasureOP, const char* inD
}
```

3. Detail implementation

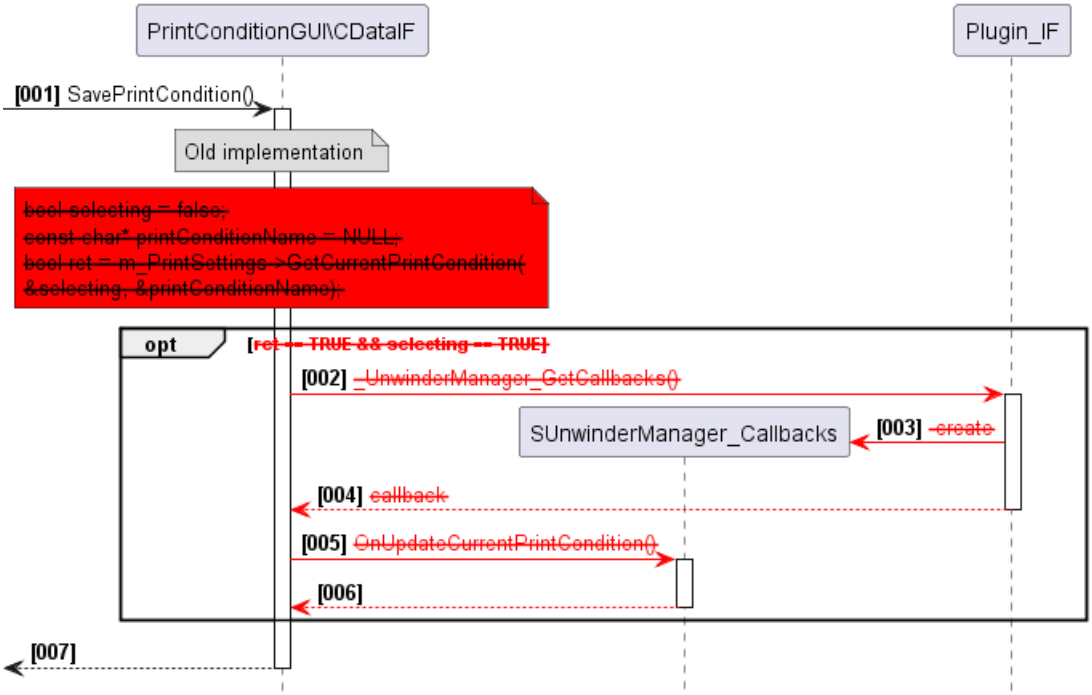
400.1 Remove CommandResource for Order2



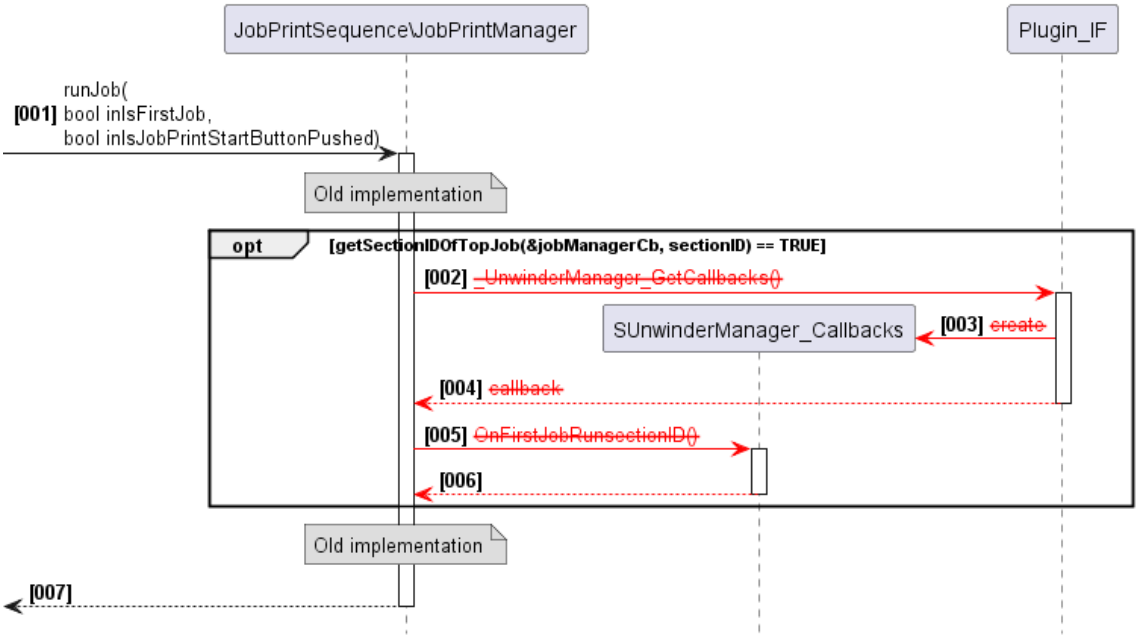
400.2 Remove channel registration when switching current print condition

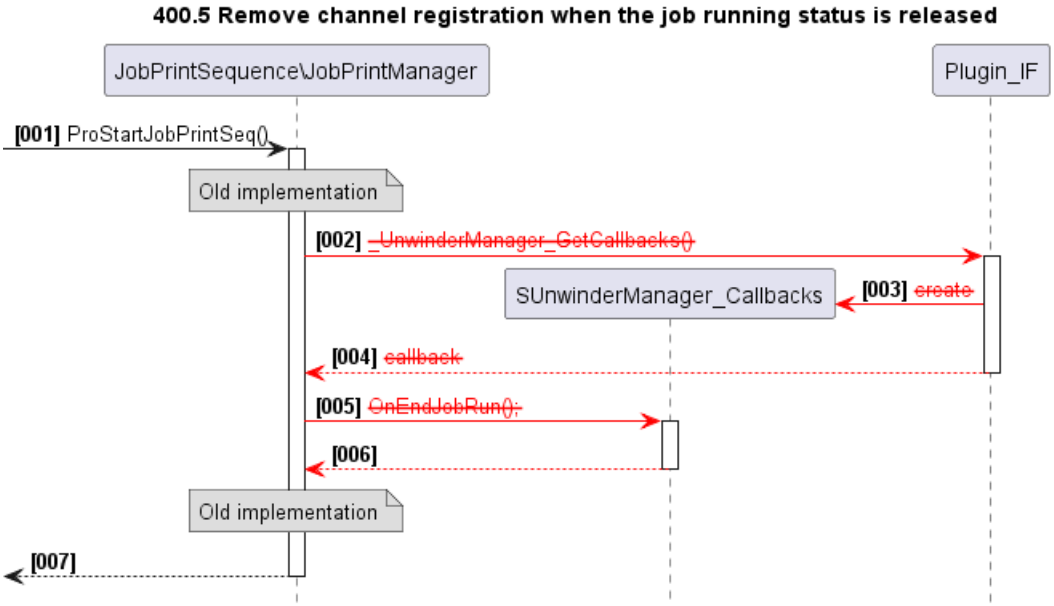


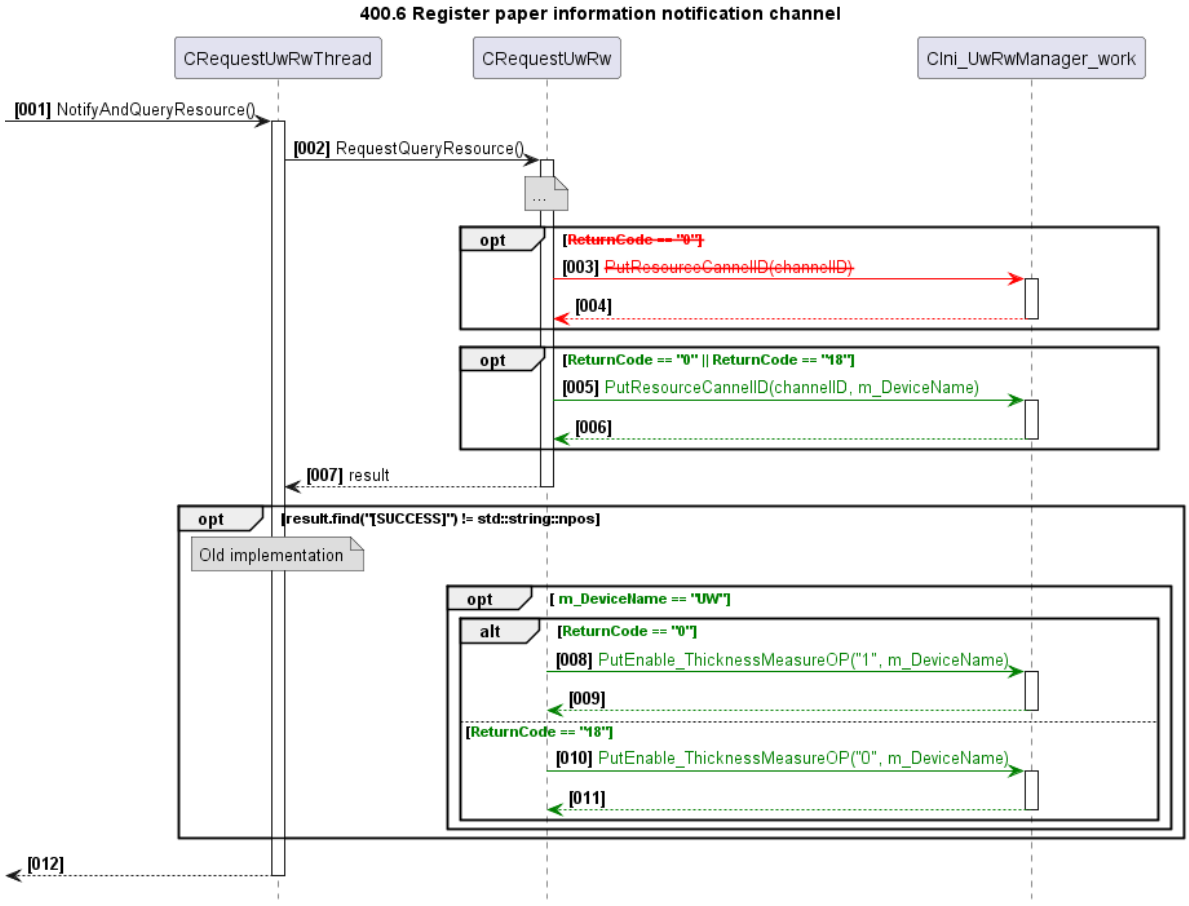
400.3 Remove channel registration when changing the current print condition setting



400.4 Remove channel registration when job is running







401. Paper information notification channel(channel deletion)

1. Description

401-01.

Deletion of the registered channel is performed by notifying the UW/RW of the “StopPresChParam” command from the controller.

- Specify the channel ID at the time of channel registration to “ChannelID” of the

"StopPresChParam" command
- Obtain the channel ID when registering a channel from "Status_CannelID" in "TP-UWRW_Communication_work.ini"
- When receiving a successful channel deletion response, also delete "Status_CannelID" in "TP-UWRW_Communication_work.ini" (store a empty value).

401-02.

When registering a channel, if the channel information has already been registered, delete the channel first.

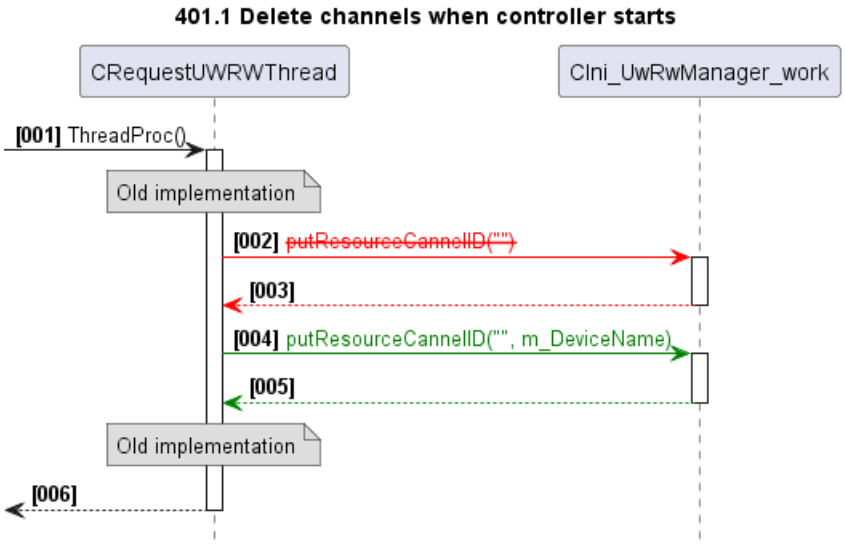
- Applicable when there is a value in "Status_CannelID" in "TP-UWRW_Communication_work.ini"

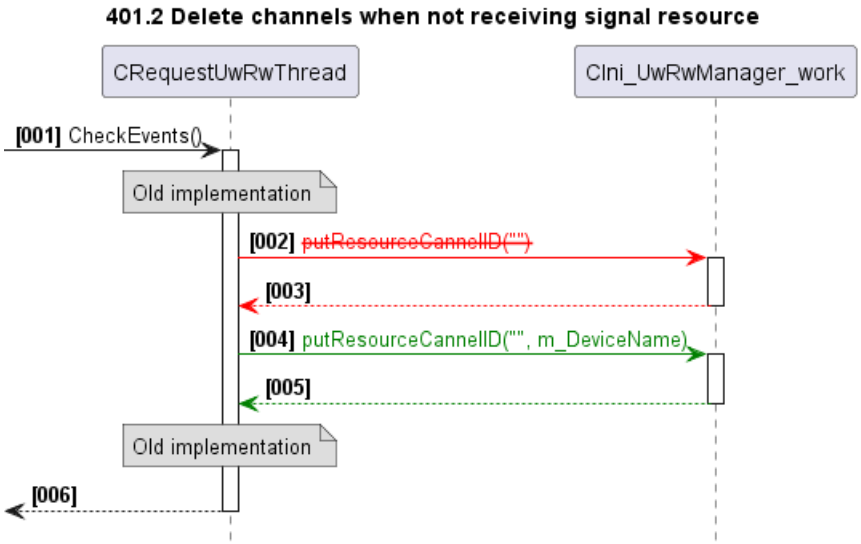
401-03. When the controller ends, delete the channel first.

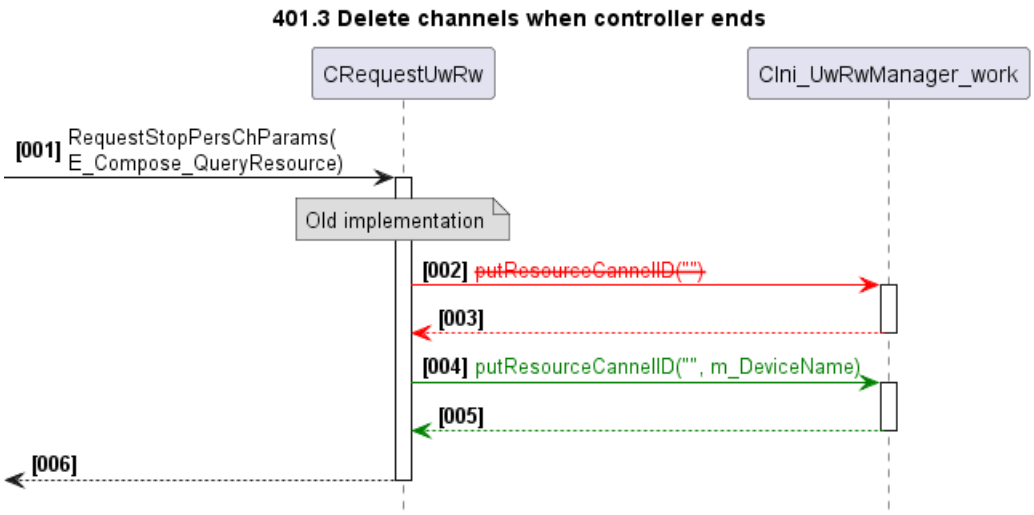
2. Solution

- In CRequestUwRwThread::ThreadProc(), CRequestUwRwThread::CheckEvents(), and CRequestUwRw::RequestStopPersChParams(), update the call to CIni_UwRwManager_work::putResourceCannelID() with DeviceName.

3. Detail implementation







402. Paper information notification channel (receiving periodic notifications)

1. Description

402-02.

Receive the following information with the “SignalResouce” command from UW/RW, store necessary information in “TP-UWRW_Communication_work.ini”

- Receive but do not need to store
  - Resource/Dimension.X (UW/RW paper width)
  - Resource/MediaType (UW/RW media name)
  - Resource/Weight (UW/RW paper weight)
  - Resource/Thickness (UW/RW paper thickness)
  - Resource/UWDancerWebTension (UW dancer web tension)
  - Resource/RWDancerWebTension (RW dancer web tension)
  - Resource/RWWindingTension (RW winding tension)
- Items that need to be stored
  - Resource/Dimension.Y (for UW: remaining amount of paper, for RW: windable amount of paper)
  - Resource/scr:RollDiameter (UW/RW roll diameter)
  - Resource/scr:StopRollDiameter (UW/RW stop roll diameter)

402-03.

Regarding 402-02, for the information that the value of “-1” was notified in “SignalResouce” from UW/RW, do not store the received information in “TP-UWRW_Communication_work.ini”.

- ① UW/RW hides the remaining amount of paper on the operation panel when the roll is not set, and notifies the TP of -1 using “SignalResouce”.
- ② UW/RW calculates the approximate amount of remaining paper immediately after the roll is set or the power is turned on, and displays it on the operation panel. At this time, the TP is notified to -1 by ‘SignalResource”.
- ③ When roll transport is performed, the final value for the remaining amount of paper is calculated, the display on the operation panel is updated, and the final value for the remaining amount of paper is notified to the TP by “SignalResource”.

402-04.

If there is no “SignalResouce” from the UW/RW after the interval + α seconds specified when the controller registers the status monitoring channel, it determines that the UW/RW has not started.

- +α seconds is defined in the “TP-UWRW_Communication.ini” file.
- At this time, the following warning message dialog is displayed.
  - For UW
    - (Ja) UWとの通信エラーが発生しました。
    - (En) A communication error with UW has occurred.
  - For RW
    - (Ja) RWとの通信エラーが発生しました。
    - (En) A communication error with RW has occurred.
  - At this time, a Ping confirmation timer is issued and a reconnection request is issued.

    If the ping passes, the channel registration for the status monitoring channel is performed until the channel registration is completed.

2. Solution

- In method CIni_UwRwManager_work::putPaperRollDiameter(), CIni_UwRwManager_work::putPaperRemainingAmount(), add inDeviceName parameter.
- Add method CIni_UwRwManager_work::putPaperStopRollDiameter() to save StopRollDiameter information.

Ini_UWRWManager_work.h

```
// before
class CIni_UwRwManager_work
{
    BOOL putPaperRollDiameter( const std::string& inRollDiameter );
    BOOL putPaperRemainingAmount( const std::string& inRemainingAmount );
}

// after
class CIni_UwRwManager_work
{
    BOOL putPaperRollDiameter( const std::string& inRollDiameter, const char* inDeviceName );
    BOOL putPaperRemainingAmount( const std::string& inRemainingAmount, const char* inDeviceName );
    BOOL putPaperStopRollDiameter( const std::string& inRollDiameter, const char* inDeviceName );
}
```

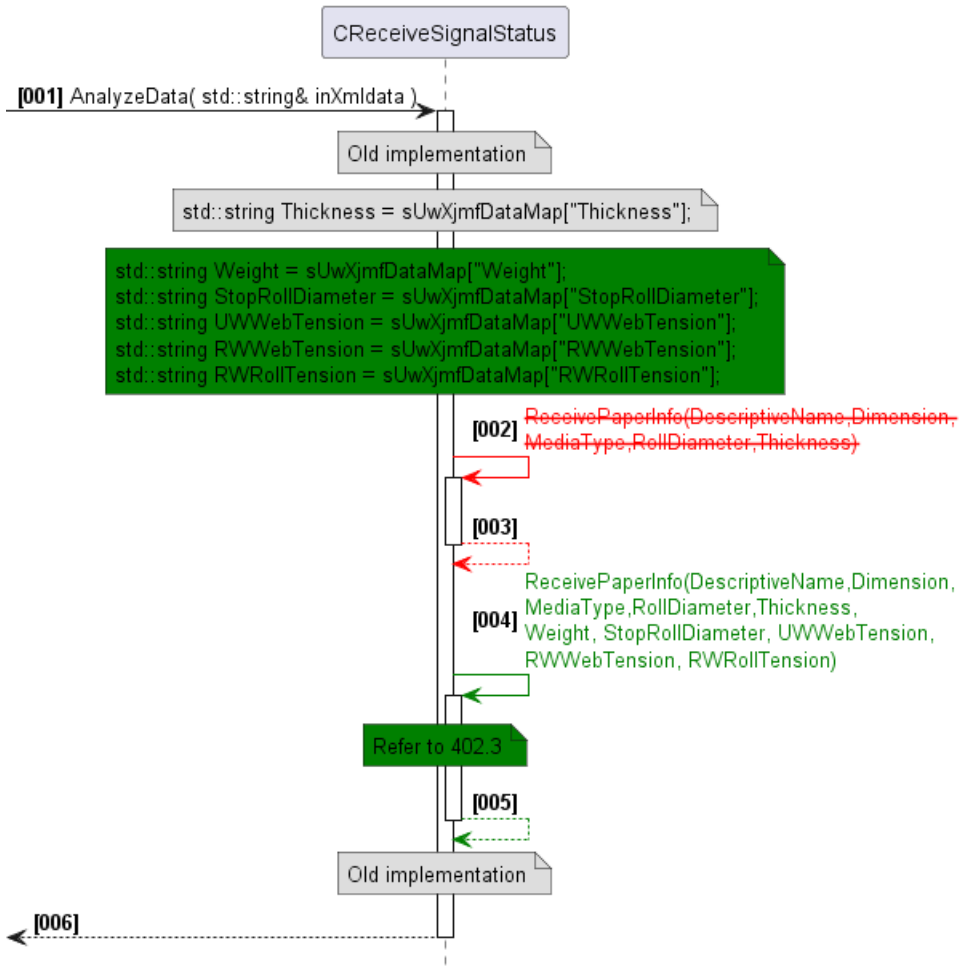
- Update method CXmlParse::ParseSignalStatus() to put more information to map: Weight, StopRollDiameter, UWWebTension, RWWebTension, RWRollTension.
- Update method CReceiveSignalStatus::AnalyzeData() with above additional information.
- In method CReceiveSignalStatus::ReceivePaperInfo(), add parameters for above additional information.
- In method CReceiveSignalStatus::ReceivePaperInfo(), if values are -1, do not store them to ini file.

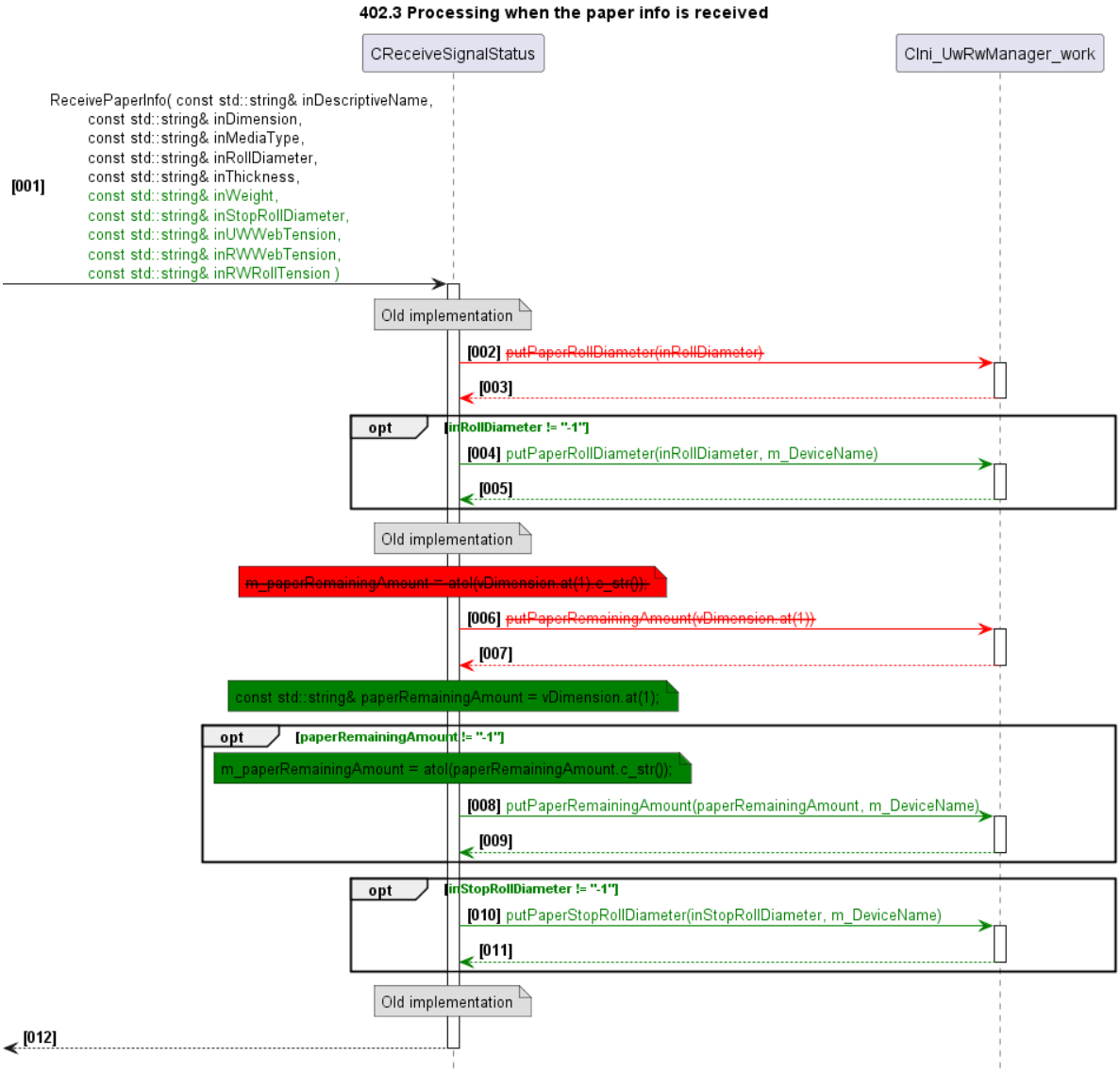
ReceiveSignalStatus.h

```
// before
class CReceiveSignalStatus
{
    BOOL ReceivePaperInfo( const std::string& inDescriptiveName,
        const std::string& inDimension,
        const std::string& inMediaType,
        const std::string& inRollDiameter,
        const std::string& inThickness )
}

// after
class CReceiveSignalStatus
{
    BOOL ReceivePaperInfo( const std::string& inDescriptiveName,
        const std::string& inDimension,
        const std::string& inMediaType,
        const std::string& inRollDiameter,
        const std::string& inThickness,
        const std::string& inWeight,
        const std::string& inStopRollDiameter,
        const std::string& inUWWebTension,
        const std::string& inRWWebTension,
        const std::string& inRWRollTension )
}
```

3. Detail implementation

402.2 Analyze signal status info notified from UWandRW_Receiver




402.3 Processing when the paper info is received
CReceiveSignalStatus
CIni_UwRwManager_work
ReceivePaperInfo(const std::string& inDescriptiveName,
const std::string& inDimension,
const std::string& inMediaType,
const std::string& inRollDiameter,
const std::string& inThickness,
const std::string& inWeight,
const std::string& inStopRollDiameter,
const std::string& inUWWebTension,
const std::string& inRWWebTension,
const std::string& inRWRollTension)
[001]
Old implementation
[002] putPaperRollDiameter(inRollDiameter)
[003]
opt
[inRollDiameter != "-1"]
[004] putPaperRollDiameter(inRollDiameter, m_DeviceName)
[005]
Old implementation
m_paperRemainingAmount = atol(vDimension.at(1).c_str());
[006] putPaperRemainingAmount(vDimension.at(1))
[007]
const std::string& paperRemainingAmount = vDimension.at(1);
opt
[paperRemainingAmount != "-1"]
m_paperRemainingAmount = atol(paperRemainingAmount.c_str());
[008] putPaperRemainingAmount(paperRemainingAmount, m_DeviceName)
[009]
opt
[inStopRollDiameter != "-1"]
[010] putPaperStopRollDiameter(inStopRollDiameter, m_DeviceName)
[011]
Old implementation
[012]