# **APP模块**：

int main(int argc, char \*argv[]) 🡪app/avm3d\_qt/qtproject/source/main.cpp

­­SVPSingleton<AppMgr>::GetInstance()->init();

void AppMgr::init() 🡪app/avm3d\_qt/source/AppMgr.cpp

SVPSingleton<AvmMsgHandler>::GetInstance()->init();

void AvmMsgHandler::init() 🡪app/avm3d\_qt/source/avm/AvmMsgHandler.cpp

g\_thread\_new("workThread", workThread, this);

m\_pAvmMsgQueue 的作用?

#ifdef DEBUG\_APP\_MSG 🡨进入AppDebugThread线程

m\_pAppDbg->Init();

#endif

gpointer AvmMsgHandler::workThread(gpointer data)

class RemoteMsgManager : public GDBus\_SuperClient

class GDBus\_SuperClient:public GDBus\_SClient

GDBus\_SClient 类需要了解下

getOptionValue()函数

void RemoteMsgManager::SyncSendData(int cmdid, SVPUint8 \*data, int len)

pThis->messagesHandler();

AvmPresent::GetInstance()->m\_pTmTask->TaskManager();

void AvmMsgHandler::messagesHandler()

{

AvmPresent::GetInstance()->MessagesHandler();

SVPSingleton<CalibPresent>::GetInstance()->MessagesHandler();

#ifdef APS\_SLOT\_SEARCH

SVPSingleton<ApaPresent>::GetInstance()->MessagesHandler();

#endif

}

void SITimerTask::TaskManager()

m\_TaskList 类貌似很重要

void AvmMsgHandler::messagesHandler()

AvmPresent::GetInstance()->MessagesHandler();

case E\_HMI\_SOFTKEY\_DYNAMIC\_CALIB:

return "Dyanmic calib soft key"; 动力校准？

case E\_HMI\_SOFTKEY\_MMI:

return "MMI soft key";

std::string toString(ENUM\_SELECTED param)

{

switch (param)

{

case E\_SELECT1:

return "selected 1 file";

case E\_SELECT2:

return "selected 2 file";

case E\_SELECT3:

return "selected 3 file";

case E\_SELECT4:

return "selected 4 file";

case E\_SELECT5:

return "selected 5 file";

case E\_SELECT6:

return "selected 6 file";

case E\_SELECT7:

return "selected 7 file";

case E\_SELECT8:

return "selected 8 file";

case E\_SELECT9:

return "selected 9 file";

default:

return "[ErrValue]";

}

}

干嘛用的？

void AvmPresent::MessagesHandler()

{

RdlMessagesHandler();

🡪 APP\_MESSAGE\_PKG 🡪 ENUM\_RDL\_MESSAGE

AppMessagesHandler();

🡪 APP\_MESSAGE\_PKG 🡪 APP\_MESSAGE\_ID

m\_pAvmModel->ViewTransitionHandler();

GetState()->Polling();

}

gpointer AppDebugMode::AppDebugThread(gpointer data)

串口调试线程

void AvmMsgHandler::init() 进来的

int AvmPresent::SetState(AvmState \*state, ENTER\_STATE\_PARAM paramEnter, EXIT\_STATE\_PARAM paramExit)

m\_pState->DoExit(paramExit);

int AvmPresent::SetState(AvmState \*state, ENTER\_STATE\_PARAM paramEnter, EXIT\_STATE\_PARAM paramExit)

{

if(m\_pState->m\_pPreState == m\_pStateIdle

&& (m\_pState == m\_pStateReverse

|| m\_pState == m\_pStateAvmOn

|| m\_pState == m\_pStateTurnLight

|| m\_pState == m\_pStateTurnLightHi

|| m\_pState == m\_pStateRadar

|| m\_pState == m\_pStateApa))

{

bEnterAVM = true;

SetTrigAvmState(m\_pState);

}

m\_pState->DoEnter(paramEnter);

}