

功能说明

项目名称: [卫星导航软件接收机 \(SatRecv\)](#)

当前版本: [2.1.1](#)

最后修改时间: [<2014-01-15>](#)

作者联系方式: [<金天 \(jintian@buaa.edu.cn\)>](#)

北京航空航天大学

修订记录

修订版本号	修订日期	修订描述	作者
V2.0.1	2011-05-01	建立基本文件	金天
V2.0.2	2011-10-15	增加组合导航部分内容	金天
V2.1.1	2014-01-15	增加多卫星导航系统支持	金天

目 录

1	编写目的	5
1.1	本文的读者群.....	5
1.2	专门术语及缩略词定义.....	5
2	子系统概述	7
2.1	当前子系统及其局限.....	7
2.2	故障处理.....	8
3	功能需求规范	9
3.1	子系统功能.....	9
3.1.1	总体结构.....	9
3.1.2	全局变量.....	10
3.1.2.1	RecvGlobal.....	11
3.1.2.2	RecvResult.....	11
3.1.2.3	RecvChannel.....	14
3.1.2.4	RecvConfig.....	17
3.1.3	局部变量.....	17
3.1.3.1	局域默认参数配置.....	17
3.1.3.2	局域自定义参数配置.....	17
3.1.4	版本管理.....	18
3.1.5	模块设计.....	18
3.2	二次开发接口.....	19
3.2.1	公共函数接口定义.....	19
3.2.2	DLL 模块接口定义.....	20
3.2.3	DLLTrack 模块接口定义.....	22
3.3	用户界面.....	24
3.3.1	控制台界面.....	25
3.3.2	可扩展界面.....	25
3.4	DSP 接口变量.....	26
3.4.1	配置参数.....	26
3.4.2	AcqConfig.....	26
3.4.3	AcqResult.....	27
3.4.4	NCOCControl.....	27
3.4.5	NCOResult.....	28
4	运行和环境要求	29
4.1	硬件	29
4.2	软件	29
4.3	网络	29
4.4	通讯协议.....	29
4.5	安装和运行.....	29
4.5.1	安装方法.....	29

4.5.2	程序命令行使用方法.....	29
5	主要模块清单	30

1 编写目的

本文介绍了通用卫星导航软件接收机的平台构架的设计实现方法。

1.1 本文的读者群

开发人员

1.2 专门术语及缩略词定义

AGC	Automatic Gain Control	自动增益控制
ASIC	Application-Specific Integrated Circuit	专用集成电路
BPSK	Binary Phase-Shift Keying	二进制相移键控
CDMA	Code-Division Multiple Access	码分多址技术
DAB	Digital Acquisition Board	
DFT	Discrete Fourier Transform	离散傅立叶变换
DLL	Delay Lock Loop	延迟锁定环
	Used to synchronize the locally generated code replica	
FFT	Fast Fourier Transform.	快速傅立叶变换
FLL	Frequency Lock Loop	锁频环路
GPS	Global Positioning System	全球定位系统
HOW	Handover word	
	17 bit truncated version of TOW	
IF	Intermediate Frequency	中频
LO	Local Oscillator	
L1	The GPS frequency band	GPS 信号频率之一 (1575.42 MHz)
L2	The GPS frequency band	GPS 信号频率之一 (1227.6 MHz)
LSB	Least significant bits	最低有效位
MSB	Most significant bits	最高有效位
NCO	Numerically Controlled Oscillator	
PLL	Phase Lock Loop	锁相环路
	Used to synchronize the locally generated carrier replica	
PRN	Pseudo Random Noise	伪随机码
RF	Radio Frequency	射频
SPS	Standard Positioning Service	
	Position calculations based on C/A code signals	标准定位服务
SVN	Satellite Vehicle Number	
TLM	Telemetry word	遥测字

	8 bit preamble used to synchronize the navigation messages	
	Time Of Week	
TOW	The GPS week starts Saturday midnight	
UHF	Ultra High Frequency	超高频
UTC	Coordinated Universal Time	协调世界时
VCO	Voltage Controlled Oscillator	压控振荡器

2 子系统概述

2.1 当前子系统及其局限

现有的软件接收机（GPSRecv）核心部分于 2006 年年底开发，主要针对 GPS 软件接收机进行的设计实现，通过 4~5 年的实际实验验证，已经具备了常用接收机的基本功能。但总体而言还存在以下几个方面的问题：

a) 总体结构基于标准 GPS 接收机设计，缺乏扩展性

现有软件接收机的设计基于一个标准的 GPS 信号处理流程，在设计的过程中没有考虑高灵敏度、高动态、多系统等一些扩展应用可能需要的功能。从而导致在其后设计的高灵敏度、多系统接收机都需要对程序结构进行修改以后才能够正常运行，且修改的版本无法融入到现有的接收机代码中。

b) 接收机的一些核心模块和参数无法修改

由于在现有版本中的部分模块和流程已经在体系结构中固定，对于一些扩展的应用无法完成对单独功能的修改。完成代码修改的人必须对整个系统结构和一部分代码完全了解以后才能够开展。由于入门时间较长，对软件接收机的修改需要较长的时间，导致一些研究进展相对困难。

c) 缺少完全自定义的可配置性

目前系统的可配置主要由核心框架读取和存储，各个扩展模块缺少自己的配置功能。虽然各个模块可以自行设计可扩展配置功能，但这种方法会导致整个系统中有两套可配置系统，反而增加了系统复杂性。

d) 缺少严格的程序版本管理和兼容机制

目前系统采用统一的版本编号，这种方法对于统一各个模块接口比较有意义，但也引入的版本控制不统一的问题。如果当接收机结构的代码进行修改以后，所有模块需要重新编译，无法在原来的接收机继续运行。

针对上述问题，新一代的软件接收机框架将所有功能模块都统一成为处理模块结构（类似 MS DirectX 系统中的 Filter 结构），采用 C 接口和 dll 扩展的方式（MS DirectX 系统采样 COM 接口，通用性好但效率低）实现统一的调度和执行。所有模块的调度频率为接收机系统更新率，跟踪模块可以细分为子跟踪模块实现 1000Hz 的更新率。

同时，各个处理模块将统一具备可扩展配置功能和版本控制功能（提供两组公共函数），使用全局的变量在各个模块中传递参数。各个模块还可以设置自己的局部参数存储区域，实现与其他指定模块的参数传递。

2.2 故障处理

（无）

3 功能需求规范

3.1 子系统功能

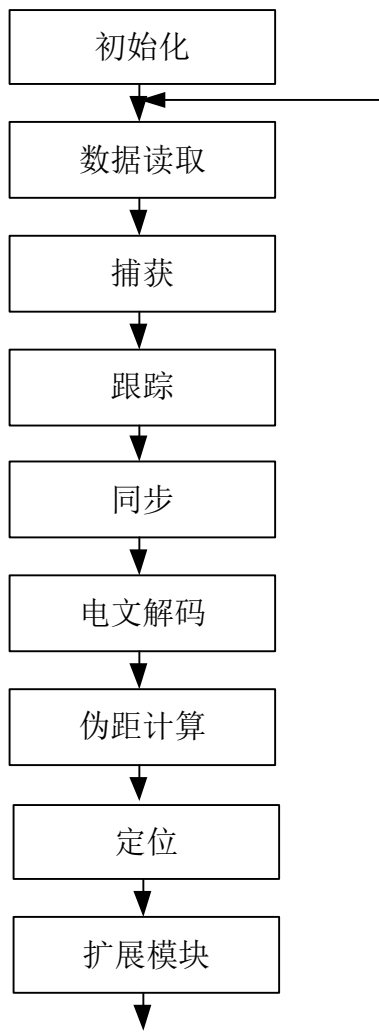
3.1.1 总体结构

软件接收机预计将包括 9 种常见的处理模块类型（初始化模块、数据采集模块、捕获模块、跟踪模块、同步模块、解码模块、伪距计算模块、定位模块和扩展模块），预计实现一个基本的卫星导航软件接收机至少需要 17 个具体的处理模块。

模块名称	功能介绍	输入	输出
初始化模块	负责对接收机进行初始化,完成对扩频码、载波和各个模块的初始化,以及全局只需要一次初始化的内容	/	扩频码
数据模块	从硬件系统或者文件系统采集数据	/	采集数据数组
捕获模块	完成信号的捕获,可以具体细分为粗略捕获和精确捕获两个部分	采集数据数组 扩频码	捕获的码相位和载波频率等
跟踪模块	完成信号的跟踪,其中相关器等更新频率为 1000Hz 的模块采用子跟踪模块实现	捕获的码相位和载波频率等	跟踪的码相位、载波相位、频率和相关结果等
同步模块	实现对相关结果的同步,具体包括位同步和帧同步	相关运算结果	同步的位置 导航电文
解码模块	根据不同卫星系统实现星历的解码	导航电文	星历和历书
伪距模块	根据跟踪状态信息和卫星信息提取伪距信息	跟踪信息、接收机和卫星位置	伪距

		信息	
定位模块	实现卫星、接收机的定位解算	卫星参数、伪距	接收机位置
扩展模块	实现用户可扩展功能	/	/

其主要流程图如下，



3.1.2 全局变量

全局变量由 struct RecvGlobal 表示，涵盖所有全局变量的结构体。具体包括全局接收机变量、全局通道变量和全局配置参数变量三个部分的变量内容。

struct RecvGlobal	涵盖所有全局变量的结构体 一个 Global 中包括一个 Config、多个
-------------------	---

	Receiver 和多个 Channel
struct RecvResult	全局接收机变量
struct RecvChannel	全局通道变量
struct RecvConfig	全局配置参数变量

3.1.2.1 RecvGlobal

```

int iVersionMajor1, iVersionMajor2;
int iVersionMinor1, iVersionMinor2;
int iSysCount, iChannelCount;
int iTick, iBlockSize;

struct RecvResult* pRecvResult; // 接收机系统数量
struct RecvChannel* pRecvChannel; // 接收机通道数量
struct RecvConfig pRecvConfig; // 配置参数数量

```

3.1.2.2 RecvResult

pRecvResult[iSysCount]表示多个系统的接收机状态，有关公共状态信息放入 global 结构体中。

使用枚举类型 enum SATFORMAT 表示卫星导航系统的类别，使用结构体 struct NAVFORMAT 表示卫星导航信号体制。

主要信号体制参数包括：

iSatCodeLength	扩频码码长（例如 1023）
iSatCodeRate	扩频码速率 (x1.023Mcps, 例如 2; 为 0 表示特殊码速率)
iSubCodeRate	扩频码子载波倍数（例如 14）
iSecCodeLength	二级码长度（例如 20）
iPilot	是否为导频通道（1/0）

一些目前常用的信号体制包括：

信号类型	名称	调制方式	码长	码速 率	子载 波	二级	导 频
GPS_L1_CA	GPS L1 C/A	BPSK	1023	1	0	0	0

GPS_L1_P	GPS L1 P	BPSK	0	10	0	0	0
GPS_L1_C_data	GPS L1 C data	BOC	10230	1	1	0	0
GPS_L1_C_pilot	GPS L1 C pilot	TMBOC	10230	1	1	1800	1
GPS_L1_M	GPS L1 M	BOC	1023	5	10	0	0
GPS_L2_P	GPS L2 P	BPSK	0	10	0	0	0
GPS_L2_C_data	GPS L2 C	BPSK	10230	1	0	0	0
GPS_L2_C_pilot	GPS L2 C	BPSK	767250	1	0	0	1
GPS_L2_M	GPS L2 M	BOC	1023	5	10	0	0
GPS_L5C_data	GPS L5 C data	BPSK	10230	10	0	10	0
GPS_L5C_pilot	GPS L5 C pilot	BPSK	10230	10	0	20	1
GLONASS_G1	GLONASS G1	BPSK	511	0	0	0	0
GLONASS_G2	GLONASS G2	BPSK	511	0	0	0	0
GLONASS_G5	GLONASS G5	BPSK	1023	1	0	0	0
Galileo_E1_OS_data	Galileo E1OS data	MBOC	4092	1	1	0	0
Galileo_E1_OS_pilot	Galileo E1OS pilot	MBOC	4092	1	1	25	1
Galileo_E1_PRS	Galileo E1PRS	BOC	4092	0	15	0	0
Galileo_E5a_data	Galileo E5a data	AltBOC	10230	10	0	20	0
Galileo_E5a_pilot	Galileo E5a pilot	AltBOC	10230	10	0	100	1
Galileo_E5b_data	Galileo E5b data	AltBOC	10230	10	0	4	0
Galileo_E5b_pilot	Galileo E5b pilot	AltBOC	10230	10	0	100	1
Galileo_E6CS_data	Galileo E6CS data	BPSK	5115	5	0	0	0
Galileo_E6CS_pilot	Galileo E6CS pilot	BPSK	5115	5	0	100	1
Galileo_E6PRS	Galileo E6PRS	BOC	5115	5	10	0	0
Beidou2_B1I	Beidou2 B1I	BPSK	2046	2	0	20	0
Beidou2_B1C_data	Beidou2 B1 C data	BOC	10230	1	1	0	0
Beidou2_B1C_pilot	Beidou2 B1 C pilot	TMBOC	10230	1	1	0	1
Beidou2_B1A_data	Beidou2 B1A data	TDDM	4092	2	14	0	0
Beidou2_B1A_pilot	Beidou2 B1A pilot	TDDM	4092	2	14	0	1
Beidou2_B2a_data	Beidou2 B2a data	TDAltBOC	10230	10	0	20	0
Beidou2_B2a_pilot	Beidou2 B2a pilot	TDAltBOC	10230	10	0	20	1
Beidou2_B2b_data	Beidou2 B2b data	TDAltBOC	10230	10	0	10	0
Beidou2_B2b_pilot	Beidou2 B2b pilot	TDAltBOC	10230	10	0	10	1
Beidou2_B3A_data	Beidou2 B3A data	BPSK	10230	10	0	0	0
Beidou2_B3A_pilot	Beidou2 B3A pilot	BPSK	10230	10	0	0	1
Beidou2_B3C_data	Beidou2 B3C data	TDDM	51150	5	0	0	0
Beidou2_B3C_pilot	Beidou2 B3C pilot	TDDM	51150	5	0	0	1

使用 `char* data_packed`; `char* data_unpacked` 分别支持多通道的数据采集功能。

使用 `char** code_packed`; `char** code_unpacked`, 分别支持多个卫星导航系统的不同卫星的扩频码。

具体通道变量包括:

类别	类型	变量名	涵义
接收机类型	enum SATFORMAT	iFormatID	通道类型
接收机	struct NAVFORMAT	sNavFormat	接收机信号体制
卫星数量	int	iSatNumber	卫星数量
量与扩	char*	pSatCode	扩频码
频码	char*	pSecCode	二级扩频码
中频信号参数	int	iSample	中频采样率 (Hz)
	int	iDataIF	中频频率 (Hz)
	int	iDataBit	采样位数 (bit)
中频信号	char*	pDataPacked	压缩存储的中频数据
	char*	pDataUnPacked	非压缩存储的中频数据
卫星情况	int	nSVs	通道中用于跟踪的卫星
	int	nNavs	通道中用于定位的卫星
接收机定位结果	double	fllh[3]	纬度、经度、高度 (度度米)
	double	fxyz[3]	ECEF 坐标 (米)
	double	fVel[3]	ECEF 速度 (米/秒)
	double	fSpd	地面速度 (米/秒)
	double	fHdg	航向角 (度)
	double	fRoc	爬升角 (度)
	double	fGDOP	几何分布因子
	double	fClockErr	时钟误差 (米)
	double	fOscErr	晶振误差 (ppm)
	bool	bValid	定位结果有效
接收机时间信息	int	dYear, dMon, dDay,	年月日
	int	dHour, dMinute	时分
	double	fSecond	秒
	int	dWN,	星期计数和星期内日计数

		dDayOfWeek	
	double	fWeekMSec, fLocalMSec	星期内 HOW 毫秒和本地毫秒
星历和 历书	struct DecEphemeris*	eph	星历
	struct DecEphemeris*	new_eph	临时星历
	struct DecAlamanc	Alm	历书
卫星位 置预测	int	iSatTick[32]	卫星预测时刻
	double	fSatPos[32][3]	卫星位置
	double	fSatVel[32][3]	卫星速度
	double	fSatSDT[32]	卫星钟差
	double	fSatElv[32]	卫星仰角
	double	fSatAzi[32]	卫星方位角
内部测 试	double	fTest[12]	内部测试数据

3.1.2.3 RecvChannel

RecvChannel[iChannelCount]表示多个通道状态。为了完成对多系统多通道的支持，在通道信息中会增加系统和频段的信息。

其余变量参考以前的 Channel 配置，具体通道变量包括：

类别	类型	变量名	涵义
卫星 与通 道	enum	iFormatID	通道类型
	SATFORMAT		
	int	iRecvIndex	通道对应接收机编号
	int	iSVN	卫星编号
跟踪 信息	double	fAcqFreq	捕获多普勒频率 (Hz)
	double	fDoppler	跟踪多普勒频率 (Hz)
	double	fPrevCAPhase	前一个跟踪码相位 (Sample)
	double	fTrackCAPhase	跟踪码相位 (Sample)

	double	fCNO	载噪比 (dBHz)
	double	fSNR	信噪比 (倍数)
	double	fIFPhase	载波相位小数部分 (Cycle)
	double	fIFFullPhase	全载波相位 (Cycle)
状态	bool	bAcqStatus	是否捕获
	bool	bReAcqStatus	是否重新捕获
	bool	bTrackStatus	是否跟踪
	bool	bBitSyncStatus	是否比特同步
	bool	bPreambleStatus	是否帧同步
	bool	bNavStatus	是否用于定位
统计 计数	unsigned int	iBitLostCount	比特丢失计数
	unsigned int	iNavDataCount	导航电文解码计数
	unsigned int	iHiLockCount	有效跟踪计数
数据 与电 文	int	iCrossingPos	跳变点位置
	int	iPreamblePos	同步帧位置
	int	bPreambleMode	帧同步模式
	int	iPreambleHOW	HOW 计数
	int	iTrackDataPos	I 支路结果计数
	int	iTrackData[300*5*20]	I 支路结果
	int	iNavDataPos	导航电文结果计数
	char	iNavData[300*5]	导航电文计数
	char	iSF	子帧状态
	int	iStatusCode	内部状态 (调试使用)
	int	aPos[20]	直方图同步 (未来不使用)
	相关 器	double	fCorrelatorCPhase;
double		fCorrelatorDoppler	相关器使用的多普勒
double		fCorrelatorIFPhase	相关器使用的载波相位
伪距	double	fTAO	伪距 (米)
相关	double	fCorrectedTAO	伪距修正量 (时钟、大气)

	double	fTropo	对流层修正
	double	fIono	电离层修正
卫星 信息	double	fAzimuth	方位角（度）
	double	fElevation	仰角（度）
	double	SVP[3]	卫星位置（ECEF，米）
	double	SVV[3]	卫星速度（ECEF，m/s）
	double	SDT	卫星时钟修正（米）
相关 器结 果	bool	bCorrelatorDumpReady	相关器此时是否有输出结果
	int	iCorrelatorLen	相关器相干积分长度
	int	iDumpIE iDumpIP iDumpIL iDumpQE iDumpQP iDumpQL	相关器 Dump 的积分结果
	int	iAccumIE iAccumIP iAccumIL iAccumQE iAccumQP iAccumQL	相关器 Accumulate 的积分结果
	double	sumI2, sumQ2	相关器累加结果
	double	sumI2I2, sumQ2Q2	相关器能量累加结果
跟踪 环路	double	fCodeDiscri0	码环路储能
	double	fPhaseDiscri0	载波环路储能
	double	fCodeDiscri	码环路鉴别器输出（chip）
	double	fPhaseDiscri	载波环路鉴别器输出（周）
环路 指示	double	fPLLLock	PLL 锁定指示器
	double	fFLLLock	FLL 锁定指示器

器			
子载波	double	fSubDiscri0	子载波储能原件
	double	fSubDiscri	子载波鉴别器
	double	fSubCodePhase	子载波码相位
	double	fSubCarrPhase	子载波相位

3.1.2.4 RecvConfig

```

int iConfigCount;
char** saConfigID;
char** saConfigValue;
int iModuleCount;
char** saModuleName;
void** paModulePointer;

```

3.1.3 局部变量

3.1.3.1 局域默认参数配置

ParaDesc 结构体用于描述各个处理模块的局域默认参数，具体包括参数名称、参数默认值和参数描述三个部分，具体定义如下。

```

Char* sParaID;
char* sParaDefault;
char* sParaDesc;

```

其中，参数 0 表示模块自身的说明。

3.1.3.2 局域自定义参数配置

配置文件的读取由主程序完成，其他模块可以独立修改配置参数的内容，但原则上不能够新增和删除配置参数项。对于一个参数量具有多个数值的情况，以最后一个

给出的参数量为准。

同时每一个处理模块都带有一个 `void` 的指针（模块个性化指针），用于实现模块之间的交互。因此可以通过参数配置查询到其他模块或者本模块所使用的个性化指针结果。

3.1.4 版本管理

接收机的所有模块采用四个数字表示系统的版本信息（`a.b.c.d`），其中前三位表示主程序版本号。暂定 `a=2`，`b` 和 `c` 表示配置文件版本号。`D` 为 6 位数字，表示模块的最后修改日期（`YYMMDD`），用于标识文件版本信息。各个具有前三位相同版本号的模块相互兼容。

例如，`v2.0.1.110330`，表示 0.1 的配置文件结构体，模块编译日期为 11/03/30。

3.1.5 模块设计

模块至少需要具备以下 4 个接口（`GetDefault`，`Init`，`Process`，`UnInit`），分别完成接口信息查询、初始化、信号处理和退出功能。其中，`GetDefault` 函数将返回本模块功能、配置参数接口、配置参数默认数值和配置参数说明等内容；`Init` 函数将完成全局变量指针的传递工作，`Process` 函数具体实现模块信息的处理，`UnInit` 函数实现退出功能。

`GetDefault` 函数通过读取一个 `config_info` 结构体，类似 `int argc`，`char** argv`，结构体中记录了可配置参数的数量，可配置参数名称和可配置参数说明（0 为本模块、说明中使用“`ID Default Info`”空格分隔参数的字符串、默认数值和摘要说明。该函数为配置文件生成器提供参考。

`Init` 函数全局接收机变量、全局通道变量和全局配置变量的传递。其中全局接收机变量只有一个，而根据通道的多少可以设置全局通道变量中的数组的长度，全局配置变量用于指示全局可配置参数，使其他模块可以对配置信息进行修改。接收机变量使用 `result` 表示，通道变量使用 `channel` 表示，配置变量用 `config` 表示（仅传送配置文件的文本信息，不传送解码以后的信息）。各个模块也可以根据自身需求，自行设置相应的模块内全局变量。

`Process` 函数具体负责信号的处理，其信息主要来自于全局接收机变量和全局通道变量。

`Unint` 函数负责退出处理，具体需要删除本模块中新申请的内存空间，关闭记录文件等。

系统将每一个处理模块保留一个 `void` 指针用来实现多个处理模块之间的信息直接交互（该指针数据放在 `config` 结构体中）。该功能的使用需要两个处理模块之间完成沟通后，自行进行通信。

3.2 二次开发接口

二次开发接口主要分为两类：`DLL` 模块接口和 `DLLTrack` 模块接口。这两个二次开发接口模块的区别如下表所示。`SatRecv` 提供了一套公共接口文件，用于实现各个模块的基本操作。

表 1 二次开发接口描述

二次开发接口	调用频率	功能
DLL 模块	每秒 25 次（可通过 <code>global.block</code> 配置）	标准处理模块 由 <code>SatRecv.exe</code> 负责调用
DLLTrack 模块	每秒 1000 次	专用跟踪处理模块，用于跟踪模块的高更新率调用 由 <code>TrackMain</code> 模块负责调用
Common 代码	/	为所有模块提供基本公共接口函数

3.2.1 公共函数接口定义

类别	函数名	功能
导航系统	<code>GetSatFormat</code>	获取卫星导航系统名称
	<code>GetSatShortFormat</code>	获取卫星导航系统简写
	<code>GetNavFormat</code>	获取导航系统信号体制
配置参数	<code>AddConfigItem</code>	增加一个参数配置项
	<code>LoadConfigInt</code>	读取一个整型参数配置项

	LoadConfigDbf	读取一个浮点型参数配置项
	LoadConfigStr	读取一个字符串参数配置项
	GetConfigCount	获得参数配置个数
	UpdateDefault	更新默认参数配置信息
接收机通道管理	ClearChannel	清除通道信息
	ClearResult	清除接收机状态信息
	SVinChannels	查询接收机的跟踪卫星信息

3.2.2 DLL 模块接口定义

DLL 模块需要包含一个头文件 `global.h`，内部存储了有关通道和接收机状态的结构体。

DLL 模块需要实现四个接口：初始化接口（`Init`）、处理接口（`Process`）和退出接口（`UnInit`），以及一个查询接口（`GetDefault`）。其中，初始化接口 `Init` 在程序启动时调用一次，退出接口 `UnInit` 在程序退出时调用一次，处理接口 `Process` 每处理完成一次调用一次（默认每秒调用 25 次）。

DLL 模块示例程序如下所示：

```
// DLLAppDDiff.cpp : 定义DLL 应用程序的导出函数。
//
#include <stdio.h>
#include <conio.h>
#include <malloc.h>
#include <math.h>
#include <windows.h>

#include "global.h"

struct RecvGlobal *pglobal;

struct ParaDesc defaults[] =
{
    "DLLAppDDiff", "20110501", "双差测试模块",
    "ddiff.prn", "1", "查看双差的卫星编号"
};
```

```
bool APIENTRY DllMain( HMODULE hModule,
                      DWORD ul_reason_for_call,
                      LPVOID lpReserved
                      )
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

__declspec(dllexport)
int GetDefault(struct ParaDesc **pDesc)
{
    *pDesc = defaults;
    return sizeof(defaults)/sizeof(struct ParaDesc);
}

__declspec(dllexport)
bool Init(struct RecvGlobal *pGlobal)
{
    pglobal = pGlobal;

    if ((pglobal->iVersionMajor1 != VERSION1) || (pglobal->iVersionMajor2 != VERSION2) ||
        (pglobal->iVersionMajor3 != VERSION3))
    {
        fprintf(stderr, "Version unmatched (program %d.%d.%d, dll %d.%d.%d)",
                pglobal->iVersionMajor1, pglobal->iVersionMajor2, pglobal->iVersionMajor3,
                VERSION1, VERSION2, VERSION3);
        _getch();
        return false;
    }

    printf("模块 (%s, v%d.%d.%d) 正在启动。 \n", defaults[0].sParaDesc,
           pglobal->iVersionMajor1, pglobal->iVersionMajor2, pglobal->iVersionMajor3,
           defaults[0].sParaDefault);

    LoadConfigInt("ddiff.prn", &g_iPRN, -1);
}
```

```
        return true;
    }

    __declspec(dllexport)
    bool UnInit()
    {
        return true;
    }

    __declspec(dllexport)
    bool Process()
    {
        return true;
    }
}
```

3.2.3 DLLTrack 模块接口定义

DLLTrack 模块需要包含一个头文件 `global.h`，内部存储了有关通道和接收机状态的结构体。

Dump 模块需要实现四个接口：初始化接口（Init）、处理接口（Process）和退出接口（UnInit），以及一个查询接口（GetDefault）。其中，初始化接口 Init 在程序启动时调用一次，退出接口 UnInit 在程序退出时调用一次，处理接口 Process 每处理完成 1ms 数据调用一次（默认每秒调用 1000 次）。DLLTrack 模块的 Process 函数传入了一批跟踪内部状态变量，用于对跟踪状态进行分析。

Dump 模块示例程序如下所示：

```
// DLLTrackSubCorrelator.cpp : 定义DLL 应用程序的导出函数。
//
#include <stdio.h>
#include <conio.h>
#include <malloc.h>
#include <math.h>
#include <windows.h>

#include "global.h"

struct RecvGlobal *pglobal;
```

```
struct ParaDesc defaults[] =
{
    "DLLTrackSubCorrelator", "20110501", "相关器跟踪子模块",
    "track.codeoffset", "0.5", "相关器间隔",
    "track.maxcorrlen", "1", "相关器积分时间"
};

bool APIENTRY DllMain( HMODULE hModule,
                      DWORD ul_reason_for_call,
                      LPVOID lpReserved
                      )
{
    switch (ul_reason_for_call)
    {
    case DLL_PROCESS_ATTACH:
    case DLL_THREAD_ATTACH:
    case DLL_THREAD_DETACH:
    case DLL_PROCESS_DETACH:
        break;
    }
    return TRUE;
}

__declspec(dllexport)
int GetDefault(struct ParaDesc **pDesc)
{
    *pDesc = defaults;
    return sizeof(defaults)/sizeof(struct ParaDesc);
}

__declspec(dllexport)
bool Init(struct RecvGlobal *pGlobal)
{
    int i, k;
    pglobal = pGlobal;

    if ((pglobal->iVersionMajor1 != VERSION1) || (pglobal->iVersionMajor2 != VERSION2) ||
        (pglobal->iVersionMajor3 != VERSION3))
    {
        fprintf(stderr, "Version unmatched (program %d.%d.%d, dll %d.%d.%d)",
                pglobal->iVersionMajor1, pglobal->iVersionMajor2, pglobal->iVersionMajor3,
                VERSION1, VERSION2, VERSION3);
        _getch();
        return false;
    }
}
```

```
}

printf("子模块 (%s, v%d. %d. %d. %s) 正在启动。 \n", defaults[0].sParaDesc,
pglobal->iVersionMajor1, pglobal->iVersionMajor2, pglobal->iVersionMajor3,
defaults[0].sParaDefault);
LoadConfigDbl("track.codeoffset", &g_fTrackCodeOffset, -1);

return true;
}

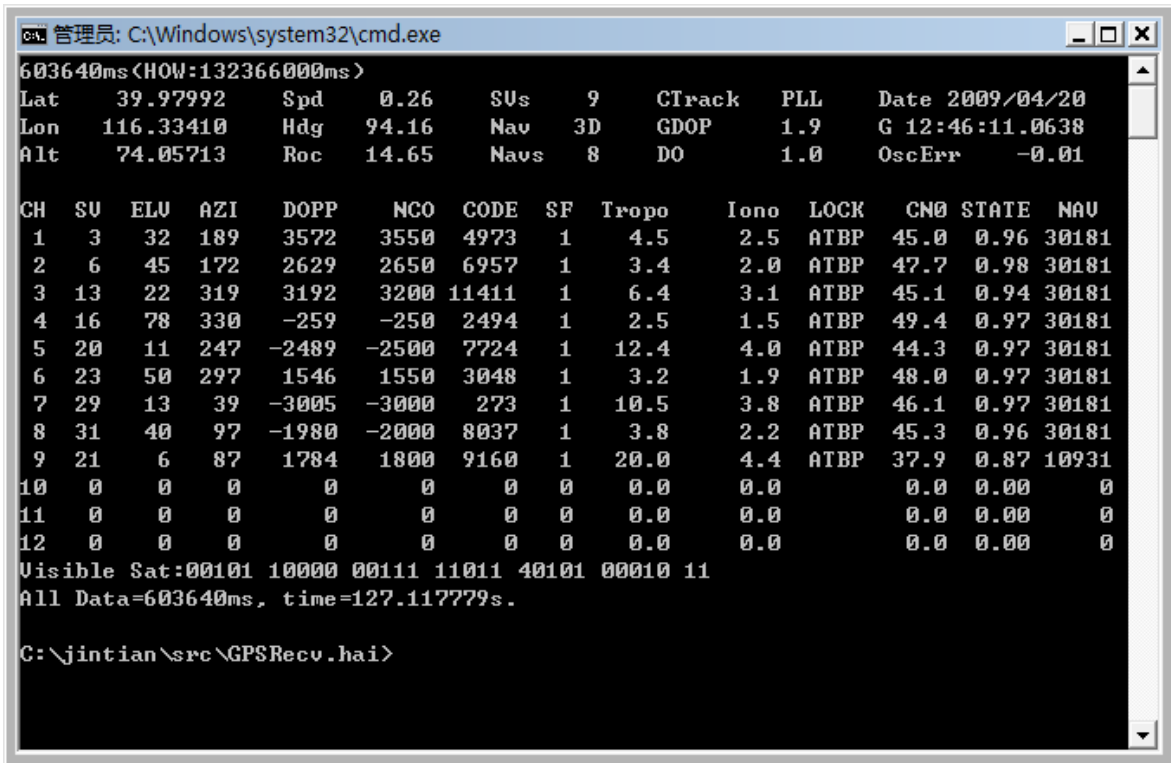
__declspec(dllexport)
bool UnInit()
{
    return true;
}

__declspec(dllexport)
bool Process(int iChannel, int nf)
{
    // iChannel表示通道编号。
    // nf表示处理数据块编号。
    return true;
}
```

3.3 用户界面

SatRecv.exe 基于控制台程序开发，可以通过扩展的模块针对用户需求实现个性化界面。

3.3.1 控制台界面



3.3.2 可扩展界面



3.4 DSP 接口变量

3.4.1 配置参数

在 DSP+FPGA 平台中，接收机初始化参数在编译前在程序内预置好，Struct RecvConfig 的包含元素如下：

类型	变量名	涵义	默认值
float	g_fTrackDLLZeta1;		
float	g_fTrackDLLZeta2;		
float	g_fTrackDLLGain;		
short	g_fTrackDLLLoss;		
short	g_fTrackDLLValid;		
short	g_fTrackDLLBn1;		
short	g_fTrackDLLBn2;		
int	g_iFreqSize;		
int	g_iDoppMax;		
short	g_fTrackFLLBn1;		
short	g_fTrackFLLBn2;		
float	g_fTrackFLLZeta1;		
float	g_fTrackFLLZeta2;		
short	g_iTrackFLLOrder;		
short	g_iSyncUpper;		
short	g_iSyncLower;		
short	g_iSyncMinDataSum;		
double	g_fRaimMinCn0;		

3.4.2 AcqConfig

用于写入对 FPGA 各通道搜索中使用的参数。

类别	类型	变量名	涵义
捕获启动 参数	int	iAcqControl	捕获控制命令
	int	iAcqIF	捕获中频
	int	iAcqFreqMax	搜索频率上限
	int	iAcqFreqMin	搜索频率下限
	int	iAcqFreqStep	粗捕获步长
	int	iAcqFineFreqStep	精捕获步长
	unsigned char	cAcqFreqNum	捕获频带数
	unsigned char	cAcqFreqThreshold	捕获门限
	unsigned char	cAcqCorrNum	相干积分时长
	unsigned char	cAcqNonCorrNum	非相干累积次数

3.4.3 AcqResult

用于写入接收自 FPGA 的各通道捕获结果。

类别	类型	变量名	涵义
捕获结果	unsigned int	iSVN	卫星号
	unsigned int	iAcqCodePhase	捕获码相位
	unsigned int	iAcqFreq	捕获多普勒频率
	unsigned int	iAcqCount	捕获结果计数

3.4.4 NCOControl

用于写入对 FPGA 各跟踪通道 NCO 的调节参数。

类别	类型	变量名	涵义
NCO 控制 参数	unsigned short	iPRN	卫星号
	short	iCPhase	码相位

	unsigned int	iCarrFCW	载波控制字
	unsigned int	iCodeFCW	码相位控制字
	int	iCodeDiscri	码环鉴相输出

3.4.5 NCOResult

用于写入接收自 FPGA 的各通道 NCO 的 I/Q 支路累积结果。

类别	类型	变量名	涵义
I/Q 支路累积 输出	unsigned int	icounter	1 毫秒计数
	int	i_early	I 支路 1 毫秒累积结果
	int	i_prompt	
	int	i_late	
	int	q_early	Q 支路 1 毫秒累积结果
	int	q_prompt	
	int	q_late	

4 运行和环境要求

4.1 硬件

(暂无)

4.2 软件

操作系统：Windows XP 以上。

4.3 网络

(无)

4.4 通讯协议

(无)

4.5 安装和运行

4.5.1 安装方法

如果不使用信号采集卡，本软件为绿色软件，无需安装使用。

如果使用信号采集卡，需要为采集卡安装驱动。

4.5.2 程序命令行使用方法

SatRecv.exe: 从 config.ini 文件中读取配置文件，或者从 ini 配置目录中选择配置文件，运行卫星导航接收机。

SatRecv.exe <IniFile>: 从 IniFile 文件中读取配置文件，运行卫星导航接收机。

5 主要模块清单

类别	模块名称	功能
初始化模块	DLLInitB1L1	B1L1 双系统初始化模块
	DLLInitBeidou2B1	BD2B1 初始化模块
	DLLInitGPSL1	GPSL1 初始化模块
数据模块	DLLDataFile	读取文件数据模块
	DLLDataSatUSB	采集卡读取数据模块
捕获模块	DLLAcqBeidou2B1	BD2B1 捕获模块
	DLLAcqGPSL1	GPS L1 捕获模块
解码模块	DLLDecnavBeidou2B1	BD2B1 导航电文解码模块
	DLLDecnavGPSL1	GPSL1 导航电文解码模块
伪距模块	DLLPRangeGPSL1	GPSL1 伪距测量模块
同步模块	DLLSyncBitBeidou2B1	BD2B1 数据同步模块
	DLLSyncBitGPSL1	GPSL1 数据同步模块
	DLLSyncPreambleBeidou2B1	BD2B1 帧同步模块
	DLLSyncPreambleGPSL1	GPSL1 帧同步模块
跟踪模块	DLLTrackMain	跟踪调度模块
	DLLTrackSubCorrelator	相关器跟踪子模块
	DLLTrackSubCorrelatorJ	高速相关器跟踪子模块
	DLLTrackSubDLLGPSL1	DLL 环路跟踪子模块
	DLLTrackSubPLLGPSL1	PLL 环路跟踪子模块
	DLLTrackSubNCOGPSL1	NCO 环路跟踪子模块
	DLLTrackIndicator	锁定指示器跟踪模块
定位模块	DLLPositionSatGPSL1	GPSL1 卫星位置解算模块
	DLLPositionIono	电离层修正定位模块
	DLLPositionTropo	对流层修正定位模块
	DLLPositionLS	最小二乘定位模块

	DLLPositionKalman	Kalman 滤波定位模块
	DLLPositionFinal	单位换算定位模块
扩展模块	DLLAppAttitude	姿态测量应用模块
	DLLAppDDiff	双差测试应用模块
	DLLAppDisplay	显示应用模块
	DLLAppShowChannel	DOS 显示应用模块
	DLLAppUltraTight	超紧组合应用模块