

VLBI 데이터 분석 가속화 기술개발 연구[2차]

제 출 문

국토지리정보원장 귀하

본 보고서를 「VLBI 데이터 분석 가속화 기술개발 연구[2차]」
과제의 최종보고서로 제출합니다.

2022. 12

한국천문연구원 원장 박 영 득

참여연구진

정태현	한국천문연구원	연구책임자
제도홍	한국천문연구원	책임연구원
변도영	한국천문연구원	책임연구원
김효령	한국천문연구원	책임연구원
김재현	한국천문연구원	선임연구원

자문위원

박용선	서울대학교	교수
이원규	한국표준과학연구원	책임연구원
조부승	한국과학기술정보연구원	책임연구원
최윤경	독일막스플랑크연구소	박사
K.Wajima	한국천문연구원	책임연구원

국토지리정보원

이진우	위치기준과	과장
박찬열	위치기준과	행정사무관
윤하수	위치기준과	감독관

VLBI 데이터 분석 가속화 기술개발 연구는 세종 우주측지관측센터 22m 전파망원경이 VLBI 관측과 상관처리, 이후 생성되는 데이터 분석을 가속화시키기 위한 시스템 성능 측정과 향상, 관측에서 상관처리에 이르는 전반의 과정을 효율화 시키는 광범위한 연구이다.

국제 측지 VLBI 서비스(IVS) 산하 국제 전파망원경들과 공동 관측을 진행하여 측지 성과를 산출하는 VLBI 관측의 특성상, 다른 전파망원경과의 시스템 호환성이 매우 중요하다. 이를 위해 세종의 백엔드 시스템인 DBBC3를 기존 2L2H에서 4L4H로 업그레이드를 진행하고 최적화 작업을 완료하였다. 세종 DBBC3의 입력 주파수 대역에 맞춘 주파수 합성기를 개발하였다. 이로써 세종은 S/X(2/8GHz) 대역과 K/Q(22/43GHz) 대역에서 4개의 광대역 VLBI 데이터 획득이 가능하게 되었다. 생성된 VLBI 데이터는 상관기를 이용하여 처리하는데, 신속한 상관처리를 위한 시간지연 오차 보정 및 데이터 저장·관리·처리를 자동화하는 절차를 정립하고 관련 유틸리티 프로그램을 개발하였다.

세종 전파망원경과 신규 컴퓨팅 노드 3기를 추가하여 성능이 향상된 리눅스 서버에 DiFX 소프트웨어 상관기를 설치하고 그 성능을 최적화 시켰다. 이로써 전파망원경 4기의 2Gbps e-VLBI 관측 및 상관처리 테스트를 성공적으로 수행하였다.

향상된 VLBI 데이터 분석 가속화 기술을 기반으로 측지·천문, 시공간 융합연구 및 우주탐사 등 활용 가능한 연구 분야를 제시하였고, 실제로 세종 우주측지관측센터와 긴밀하게 협력하여 해당 연구들을 수행중에 있다. KVN을 비롯한 4개국 7개 기관이 참여하는 동아시아 VLBI 관측망(EAVN) 유관 기관들과 EAVN 공동 관측운영 및 협력에 관한 양해각서를 2021년 체결하고, 국내외 VLBI 네트워크 협력 기반을 구축하였다. 보다 활발한 VLBI 공동관측 협력을 위해 필수적인 세종 22m 전파망원경의 시스템 성능지표 산출하였다. 본 ‘VLBI 데이터 분석 가속화 기술개발 연구[2차]’를 통하여 세종 우주측지관측센터의 VLBI 관측 및 데이터 분석 성능의 향상과 함께 국내외 VLBI 네트워크 협력을 통한 측지·천문·우주개발 등의 다양한 분야의 성과를 산출할 수 있는 기반을 구축하였다.

제1장 연구개요	1
1. 연구의 배경 및 목적	3
가. 연구의 배경	3
나. 연구의 목적	7
2. 연구 내용	8
제2장 e-VLBI 구현을 위한 DBBC 기능 고도화 및 최적화	11
1. e-VLBI 관측을 위한 DBBC3 기능 고도화	13
가. DBBC3-4L4H 업그레이드	13
나. DBBC3-4L4H 확장에 따른 입력 연결 설정 소프트웨어 개선	16
다. e-VLBI 관측에 필요한 출력 설정 정의 및 소프트웨어 인터페이스 추가	18
2. DBBC3의 측지관측 적용을 위한 VLBI 관측 소프트웨어 기능 개선	20
가. 측지 관측 적용을 위한 DBBC3 인터페이스 확장 및 개선	20
나. DBBC3 업그레이드에 따른 관측 소프트웨어 기능 개선	22
3. DBBC3 기반 VLBI 관측 안정화를 위한 성능 평가 및 최적화 수행	29
제3장 DIFX 상관처리 자동화 기술 고도화	31
1. 상관처리를 위한 표준전파원 DB 구축	33
2. 표준전파원 자동 선택·등록 유틸리티 개발	34
3. 상관처리를 위한 지연시간 보정 기능 자동화	35
가. N-포인트 방법 자동화	35
나. v2d 파일 자동 업데이트	37
4. VLBI 관측 데이터의 저장·관리·처리 자동화를 위한 관련 유틸리티 기능 업그레이드	39
가. 데이터 전송 관리 기능 유틸리티 개발	39
나. 데이터 저장 기능 유틸리티 개발	41

제4장 e-VLBI 관측 안정화를 위한 성능 평가	43
1. DiFX 상관계 성능 개선을 위한 서버 추가 연동, DiFX 상관계 SW 업데이트 등 관측 환경 조성	45
가. DiFX 서버 환경 분석	45
나. DiFX 환경 설정 및 테스트	47
2. e-VLBI 성능 업그레이드 DiFX 상관계를 이용한 e-VLBI 상관처리 테스트	47
가. 세종 + KVN 2Gbps e-VLBI 관측 설정	48
나. 세종 + KVN 2Gbps e-VLBI 상관처리를 위한 DiFX 설정	51
다. 세종 + KVN 2Gbps e-VLBI 상관처리 프린지 검출	53
3. e-VLBI 운용을 위한 사용 매뉴얼 제작 및 e-VLBI 활용 방안	56
가. e-VLBI 운용을 위한 사용 매뉴얼	56
나. e-VLBI 활용 방안	56
제5장 세종 VLBI 활용 확대를 위한 VLBI 네트워크 협력 기반 구축	59
1. KVN·EVN과 공동 관측 협력 방안	61
2. 국·내외 VLBI 관측망 참여를 위한 세종 VLBI 시스템 성능 측정 및 정량화된 성능지표 생성	64
3. 세종+KVN·EAVN 등 VLBI 관측망을 이용한 활용 분야 발굴·제시	65
가. EAVN의 측지·측성 VLBI 기준 역할 수행	65
나. 국제 K-band 측지 VLBI 관측을 통한 천체기준좌표계(CRF) 연구	66
다. 초정밀 시공간 융합연구	67
라. 국·내외 VLBI 전파망원경을 활용한 우주탐사선 정밀 위치 결정	69
제6장 광대역 e-VLBI 관측을 위한 주파수 합성기 개발 및 적용·운영방안 마련	71
1. 세종 시스템에 적합한 합성기 개발을 위한 시스템 구성도 분석, 평가	73
가. 기준신호 전송 시스템 분석, 평가	73

나. 10/100MHz 이외의 1GHz 이상의 고주파수의 기준 주파수 공급 방법 검토	75
다. 현재 설치된 22/43GHz 수신기 및 주파수 하향 변환기의 주파수 안정도 분석, 평가	77
2. 주파수 합성기 개발을 위한 요구 성능 설계	82
가. 주파수 합성기 출력 주파수 범위, 해상도, 출력 전력 계산	82
나. 22/43GHz 수신기의 관측 효율을 고려한 주파수 별 필요 위상안정도 계산	85
3. 주파수 합성기 회로 설계 및 제작	87
가. 주파수 합성기 시스템 설계	87
나. PLL회로 설계 및 제작	89
다. YIG 발진기를 활용한 RF 회로 설계 및 제작	90
라. 주파수 차폐 상자를 포함한 전체 주파수 합성기 조립	92
마. 주파수 합성기 프로그램 작성 및 주파수 합성기 완성	93
4. 주파수 합성기 종합성능 테스트	95
가. 출력 주파수 신호 크기 및 Lock 여부 측정	95
나. 출력 주파수별 위상 잡음 측정	97
다. 주요 주파수 장기 위상 안정도 측정	97
5. 세종 VLBI 시스템 적용을 위한 현장 성능 시험	100
가. 주파수 합성기 설치 및 수신기 위상 안정도 측정 setup 구성	100
나. 망원경 성능 시험을 위한 프로그램 작성	101
다. 개발된 주파수 합성기를 활용한 수신기 위상 안정도 측정	101
6. 22/43GHz 수신기 광대역화 구현을 위한 주파수 합성기 적용 방안 마련	103
7. 결론	104
참고문헌	106
용어 및 약어 정리	107

■ 목차 ■

부록	109
1. DBBC3 업그레이드 성능 시험 결과서	111
2. 세종 DiFX e-VLBI 매뉴얼	120
3. 세종 전파망원경 시스템 성능 지표	135
4. 주파수 합성기 프로그램	148
5. 주파수 합성기 점검 매뉴얼	156
6. 민간 연구자 지원 업무 협조	163
7. EAVN 협력에 관한 연구논문 출판	164
Abstract	166

<표 1-1> DBBC3 firmware 별 지원 기능	14
<표 6-1> 현재 수신기의 8-10GHz IF 주파수를 사용할 경우, 수신기 광대역화에 요구되는 주파수 합성기의 주파수 범위와 간격	82
<표 6-2> DBBC3 입력 주파수 대역인 4-15GHz IF 주파수를 사용할 경우, 수신기 광대역화에 요구되는 주파수 합성기의 주파수 범위와 간격	83
<표 6-3> 43GHz 대역 수신기의 주파수 할당 방법	84
<표 6-4> 10GHz 대역 주파수 합성기의 위상잡음	85
<표 6-5> 10-20GHz YIG 주파수 합성기의 출력 주파수 목록	88
<표 6-6> RF 회로의 주요 부품 목록	92
<표 6-7> 주파수 합성기 프로그램의 주요 기능 및 기능	93
<표 6-8> 주파수 합성기 응답 명령	94
<표 6-9> 주파수 합성기가 만족할 기본 특성	95
<표 6-10> 주파수 오프셋에 따른 1 deg rms의 위상 잡음과 13.2GHz와 17GHz에서 측정한 주파수 합성기의 위상 잡음 비교	97
<표 6-11> 안테나 고도 변화 시간	102

■ 그림목차 ■

<그림 1-1> 세종 망원경과 KVN 망원경의 위치	4
<그림 1-2> 동아시아 VLBI 관측망(EAVN) 분포	5
<그림 2-1> 세종 우주측지관측센터 DBBC3	13
<그림 2-2> DBBC3 구성도	14
<그림 2-3> DBBC3 입출력 구성	15
<그림 2-4> 입력 결합 회로 구성안	16
<그림 2-5> DBBC3 입력 선택을 위한 스위치 매트릭스 제어 창	17
<그림 2-6> Core3H의 출력 IP/Port 설정 제어 창	18
<그림 2-7> DBBC3의 Core3H 10GbE 출력 설정	19
<그림 2-8> DBBC3의 DBBC 주파수 채널 설정 GUI 창	20
<그림 2-9> Core3H의 출력 IP/Port 설정 제어 창	21
<그림 2-10> Core3H_1,2,3,4 의 입력 신호 세기 분포 표시 창	21
<그림 2-11> VLBI 관측 스크립트 실행 창	22
<그림 2-12> Mark6 상태 창 - 4개 입력 스트림 정보 표시	23
<그림 2-13> 세종 망원경을 이용한 22GHz 양편파 관측 프린지 결과 ..	24
<그림 2-14> OCT_D firmware의 동작 구성	25
<그림 2-15> OCT_D 모드의 필터 탭 설정 GUI 패널	25
<그림 2-16> 자료 처리 서버에서 Total Power Detector (VFC) 값을 이용 한 대기 흡수 계수 측정 결과	27
<그림 2-17> 업그레이드된 DSLoc 에서 기록한 시스템 온도 자료를 이용한 ANTAB 파일 결과	27
<그림 2-18> 업그레이드된 DSLoc를 이용한 Five Pointing 관측 결과 ...	28
<그림 2-18> 업그레이드된 DBBC3 모습	29
<그림 3-1> n21mh01a에 대한 clockrate 및 delay 처리결과	38
<그림 4-1> e-VLBI를 위한 클러스터와 스토리지 구성 개념도(예시)	46
<그림 4-2> 세종 DiFX belbi2 서버	46
<그림 4-3> 세종-KVN 실시간 e-VLBI 구성도	47
<그림 4-4> belbi2 서버에서 2Gbps e-VLBI 상관처리를 위한 DiFX 상관처리 초기화(화면캡처)	54

<그림 4-5> belbi2 서버에서 2Gbps e-VLBI 상관처리가 진행되며 상관처리된 visibility 데이터가 생성되는 모습(화면캡처)	54
<그림 4-6> 각 망원경에서 belbi2 DiFX 서버로 2Gbps 데이터가 손실없이 전송되는 모습(화면캡처)	54
<그림 4-7> 세종+KVN 4기 전파망원경을 이용한 2Gbps e-VLBI DiFX 상관처리 프린지 검출	55
<그림 4-8> e-VLBI를 활용 분야	57
<그림 5-1> 동아시아 VLBI 망원경 분포도	61
<그림 5-2> EAVN 공동운영 및 과학기술 협력에 관한 양해각서	63
<그림 5-3> 세종 22m VLBI 전파망원경 성능 보고서(status report)	64
<그림 5-4> EAVN K-band(22GHz) 측지 VLBI 관측 현황	65
<그림 5-5> 천체기준좌표계 성능 향상을 위한 세종+KVN+남아프리카국+호주의 전파망원경 분포	66
<그림 5-6> 전파망원경을 이용한 K-band 측지 VLBI 관측 제안서	67
<그림 5-7> 대륙간 광시계 비교를 위한 시공간 융합연구 개념도	67
<그림 5-8> 시공간 융합연구를 위한 5개 기관 양해각서 체결식	69
<그림 5-9> 유럽우주국(ESA)의 우주탐사 미션 및 JUICE 목성탐사선의 VLBI 관측 개념도	70
<그림 5-10> 우리나라의 VLBI를 활용한 우주탐사선 추적 가능 인프라 ·	70
<그림 6-1> 기준신호 전송시스템의 안테나 수신기 실 블록도.	74
<그림 6-2> 기준신호 전송시스템 1.4GHz 포트의 출력 스펙트럼	74
<그림 6-2> 1.36/1.38GHz 간섭 신호 제거를 위한 1.4GHz PLL 회로	76
<그림 6-4> 1.36/1.38GHz 간섭신호가 제거된 1.4GHz 신호 출력을 위한 PLL 회로 제작	76
<그림 6-5> 1.4GHz PLL 회로에 사용된 V1400, 1.4GHz VCO의 구성도 및 위상 잡음 측정 결과	76
<그림 6-6> 3.25GHz PDRO의 온도에 따른 위상 안정도 측정	77
<그림 6-7> 13.25GHz PDRO의 온도에 따른 위상 안정도 측정	77

그림목차

<그림 6-8> 13.25GHz PDRO의 온도에 따른 위상 안정도 측정 결과	77
<그림 6-9> 13.25GHz PDRO의 100MHz 기준 주파수 신호에 따른 위상잡음 측정 결과	78
<그림 6-10> 기존의 국부발진기를 사용한 22GHz 수신기 위상 안정도 측정 구성도	79
<그림 6-11> 22GHz 대역 수신기에 1.4GHz 하모닉 신호 인가	80
<그림 6-12> IF 100MHz 신호와 H-maser 신호의 위상 비교를 위한 측정	80
<그림 6-13> 21GHz 신호를 인가했을 때의 22GHz 대역 수신기의 온도에 따른 위상 안정도 측정 결과	81
<그림 6-14> 안테나 수신기 실의 온도 변화	86
<그림 6-15> 짧은 시간 동안의 항온항습기 동작에 따른 수신기 실 내부의 온도 변화	86
<그림 6-16> 1.4GHz와 100MHz 기준 주파수 신호를 사용한 10-20GHz YIG 주파수 합성기 구성도	87
<그림 6-17> 10-20GHz YIG 주파수 발생기 테스트 결과	88
<그림 6-18> 10-20GHz YIG 주파수 합성기의 직렬통신 포트 핀 할당과 외형	89
<그림 6-19> PLL 회로 구성도 및 제작된 PLL 회로	90
<그림 6-20> RF 부품들이 조립된 알루미늄 상자 내부	91
<그림 6-21> 주파수 합성기 차폐상자 내부와 전면 사진	93
<그림 6-22> 전력 증폭기가 추가된 주파수 합성기의 차폐 상자 내부 사진과 주파수 합성기 출력 전력	96
<그림 6-23> Marki Microwave 사의 APM-6849 증폭기의 주요 규격 ..	96
<그림 6-24> 13.2GHz와 17GHz에서의 주파수 합성기 위상 잡음 측정 결과	97
<그림 6-25> 온도 변화에 따른 위상 안정도 측정 구성도	98
<그림 6-26> 온도 변화를 위해 스티로폼 상자 안에 설치한 주파수 합성기 및 위상 비교 실험	98

<그림 6-27> 16.6GHz에서 측정된 주파수 합성기 온도에 따른 위상 안정도	99
<그림 6-28> 주파수 합성기를 22GHz 수신기 옆에 설치한 사진	100
<그림 6-29> 주파수 합성기를 22GHz 수신기의 국부발진기로 사용했을 때 위상 안정도 측정 구성	100
<그림 6-30> 안테나 고도와 수신기 실 온도 변화에 따른 22GHz 수신기의 위상 안정도 측정 결과	102

제1장

연구개요

1. 연구의 배경 목적
2. 연구 내용

1. 연구의 배경 및 목적

가. 연구의 배경

1) 측지 VLBI

한반도 주변은 점차 지각 활동이 증가하고 있으며, 특히 일본은 큰 지진이나 화산 활동이 계속 이어지고 있다. 이런 환경에서 지구적인 지각 운동 뿐만 아니라 한반도의 지각 운동의 정밀 측정은 국가안보와도 연관된 중요한 사안이다. 차량 자율 주행, 가상 현실 등 미래 산업에서도 현재보다 더 정밀한 좌표 정밀도를 요구하고 있다. VLBI는 지구 자전 및 지각의 운동을 가장 정밀하게 측정할 수 있는 요소 기술 중의 하나로 전 세계적으로 널리 사용되고 있다. 특히 측지 VLBI는 천체를 기준으로 한 천구 좌표계와 지상 좌표계를 묶어주는 유일한 수단으로 앞으로 다가오는 우주 개발 시대에도 그 중요도가 더욱 증가하고 있다.

전 세계의 전파망원경을 아우르는 측지 VLBI 관측은 2/8GHz 대역을 주로 사용하고 있으며, 점차 VGOS라는 새로 도입된 시스템 표준을 도입하여 2-14GHz 대역으로 확장되고 있다. 세종 망원경은 2/8GHz IVS(International VLBI Service for Geodesy and Astrometry) 관측에 정기적으로 참여하고 있다. 최근 세종 우주측지관측센터는 수년간 누적된 IVS 2/8GHz 측지 관측 자료를 분석하여 1mm 정도의 기준 좌표 정밀도를 달성하였다. 이 기준 좌표 값을 다른 국내 좌표 기준점으로 전파하여 사용함으로써 국내 측지계의 좌표 정밀도를 개선시킬 것으로 기대된다.

2) 국내 VLBI 현황

한국천문연구원의 전파천문본부에서는 한국우주전파관측망(Korean VLBI Network: 이하 KVN)을 개발하여 VLBI 관측연구에 매진하고 있다. 특히 세계 최초의 K(22GHz), Q(43GHz), W(86GHz), D(129GHz) 4채널 동시관측 시스템을 구축하여 운영하고 있으며, 이로부터 생성되는 8Gbps 데이터 저장, 데이터 전송 및 상관처리를 원활하게 수행하고 있다. KVN은 서울, 울산, 제주에 3개의 관측소를 운영하고 있으며, 관측소에서 기록된 데이터는 한국정보기술연구원(KISTI)에서 제공하는 10G 크레오넷망을 이용하여 대전 스토리지 서

버로 온라인 전송되고, DiFX 상관을 통하여 상관처리된 FITS 파일을 관측자들에게 제공하고 있다[1, 2]. DiFX가 수행하는 관측시간은 연간 2000 시간이며 처리하는 데이터 양도 약 4500 TB에 이른다. 이러한 방대한 양의 데이터 처리를 위하여 데이터 기록, 데이터 전송, 데이터 분류, 상관처리 등의 과정을 프로세스별로 구분하고, 각 프로세스마다 관련된 유틸리티를 개발하여 운영하고 있다. 이러한 데이터 전송 및 상관처리를 통하여 구축된 기술은 세종 측지 시스템에서 필요로 하는 VLBI 관측 데이터의 전송 및 상관분석 가속화 기술개발에 적합하다.

세종 VLBI 망원경은 KVN과 더불어 국내에서 VLBI 연구 관측을 수행할 수 있는 유일한 전파 관측 시스템이다. KVN 망원경은 서울, 울산, 제주에 위치하여 삼각형의 구성을 가지고 있는데, 세종 망원경은 삼각형의 KVN 망원경 구성의 가운데 지점에 위치하여 측지 및 천문 관측시 기준 망원경으로서 역할을 수행하기에 이상적인 위치에 놓여있다. 2016년부터 세종-KVN의 22GHz 측지 관측을 정기적으로 수행하고 있으며, 22/43GHz 를 이용한 천문 목적의 공동 관측도 수행하고 있다.



<그림 1-1> 세종 망원경과 KVN 망원경의 위치

3) 동아시아 VLBI 관측망(EAVN)을 활용한 측지 VLBI

동아시아 VLBI 관측망(East Asian VLBI Network: 이하 EAVN)은 한국의 VLBI 관측망인

KVN, 일본의 VERA, 중국의 CVN 등 3개국 21개 망원경을 연결한 최대 기선 5000km 정도의 거대 관측망으로, 미국의 초장기선 전파망원경배열(Very Long Baseline Array: VLBA), 유럽 VLBI 전파망원경 네트워크(European VLBI Network: EVN)과 함께 세계 3대 VLBI 네트워크로 운영 중이다. 특히, 최근 EAVN은 22GHz 대역의 고주파에서 측지 VLBI 관측을 수행하고 있는데, 앞으로 EAVN의 측지 VLBI 관측에 세종 망원경도 참여한다면, 한반도 VLBI 관측망 기선인 500km에서 5,000km로 확장된 기선으로 인하여 분해능이 10배 높아져 정밀 측지 VLBI 달성이 가능하다.

EAVN 관측 데이터는 한국천문연구원내에 위치한 대전상관센터에서 처리하고 있다. 데이터 전송은 이전까지 중국과 일본에서 하드디스크를 배송하는 방식을 사용하여 왔는데, 최근에 온라인 데이터 전송 방식을 활용하여 데이터 송수신을 구현하고 있다. 이러한 환경구축과 운용을 통하여 축적된 기술들을 세종 측지 시스템에 추가함으로써 세종 측지 시스템과 동아시아망 사이의 데이터 송수신이 구현된다.



<그림 1-2> 동아시아 VLBI 관측망(EAVN) (출처: Nature Astronomy). 동아시아 지역에 있는 VLBI 망원경 분포.

최근 동아시아 VLBI 망원경(EAVN)을 중심으로 호주, 러시아까지 포함한 22GHz 대역 측지 VLBI 관측이 활성화되고 있다. 세종 망원경은 전통적인 2/8GHz 측지 VLBI 관측뿐만 아니라 22/43GHz까지 가능한 망원경으로 EAVN의 측지 VLBI 관측에서 매우 중요한 위상을 가진다. 특히, EAVN 측지 VLBI 관측결과는 일본 미즈사와 망원경을 기준으로 산출되고 있는데, 2011년 후쿠시마 대지진 이후로 사실상 절대좌표의 기준점이 불안정한 상태로, 세종과 같은 안정된 기준점이 필요한 상황이다.

4) 네트워크를 활용한 VLBI 데이터 전송 및 상관처리

초고속 네트워크 기술 및 인프라의 급속한 발전은 원거리 전파망원경들을 연결하여 관측하는 VLBI 기술에도 큰 영향을 끼쳤다. 인터넷망을 통한 원격 망원경 컨트롤과 관측데이터의 전송을 비롯하여 이를 실시간으로 상관처리까지 가능하다.

KVN의 경우 서울, 울산, 제주에서 관측되는 모든 데이터를 한국과학기술정보연구원이 운영하고있는 과학기술연구망(KREONET)을 이용하여 대전 천문연구원 본원에 위치한 대전 상관센터로 전송하는 최적의 e-Transfer 방안을 개발하여 관측운영에 활용하고 있으며, 실시간 e-VLBI 상관처리에도 성공하였다. 이처럼 네트워크를 활용한 VLBI 데이터 전송 및 상관처리 효율화는 이후 자료를 분석하고 결과를 도출하는데 걸리는 시간을 크게 단축시킬 수 있다. 참고로, 앞서 언급한 것처럼 최근 EAVN의 활발한 측지/측성 VLBI 움직임과 함께 EAVN의 중심에 위치한 세종 망원경의 중요성은 더욱 높아질 것으로 예상된다.

5) VLBI 데이터 분석 가속화 기술개발 연구 1차 년도 사업성과

“VLBI 데이터 분석 가속화 기술개발 연구” 1차 년도 사업으로부터 국내외 e-VLBI 기술개발 동향을 분석하였으며, 실제 e-Transfer 효율화 및 e-VLBI 적용을 위한 세종 VLBI 시스템을 분석하고, 관련 스크립트를 개발하였다. 이를 바탕으로 KVN과의 시험 관측을 수행하고 상관처리 효율성을 테스트하였다.

그리고 VLBI 데이터 분석 가속화에 있어서 핵심 시스템인 DBBC 하드웨어를 개선하는 작업을 통하여, 세종 망원경의 DBBC와 관측 전반을 제어하는 필드시스템(Field System) 기능을 업그레이드하였다.

나. 연구의 목적

VLBI 망원경 시스템의 디지털화, 초고속 네트워크 인프라 및 소프트웨어 기반의 상관처리 시스템의 발달로 우편을 통하여 VLBI 관측 데이터를 상관센터로 ‘배송’하는 방식에서 벗어나, 초고속 네트워크를 통해 망원경에서 상관센터로 ‘전송’하는 방식이 급속히 보편화되고 있으며, 소프트웨어 상관기를 이용한 실시간 상관처리(e-VLBI)까지 가능하게 되었다.

이에 우리나라의 과학기술정보연구원에서 제공하고 있는 우수한 초고속 네트워크 인프라 (KREONET)를 기반으로 한반도 측지 VLBI 관측망의 기준인 세종 우주측지관측센터에서 국내외 측지 VLBI 관측과 상관처리 및 데이터 분석을 선도하기 위하여, e-Transfer와 e-VLBI, 상관분석 업무 효율화를 위한 스크립트를 개발하고 관련 하드웨어 개선을 진행해 오고 있다.

1차 년도에서는 VLBI 데이터 분석 가속화를 위하여 e-Transfer 관련 스크립트 개발 및 상관분석 업무의 효율화에 중점을 두었으며, 이러한 1차 년도 사업 성과를 기반으로 2차 년도에서는 상관기 성능 개선을 위한 업그레이드를 진행하고, 실시간 e-VLBI 구현을 바탕으로 VLBI 관측에서부터 상관분석 처리에 소요되는 시간을 최소화시키고자 한다. 또한 실제 운용을 위하여 매뉴얼과 사용자 교육을 진행한다. 특히, 고주파수 측지 VLBI에서 가장 오랜 역사를 가지고 있는 EAVN과 공동 관측 협력 및 활용을 위한 방안을 마련하고, DBBC 기능 고도화 및 광대역 e-VLBI 관측을 위한 주파수 합성기를 개발하고자 한다.

- VLBI 관측부터 상관분석(Correlation process) 처리에 소요되는 시간을 최소화 할 수 있는 기법 개발 및 VLBI 운영 시스템에 적용
- VLBI 데이터 분석 가속화를 위해 필요한 장비(DBBC: Digital Base Band Converter) 및 광대역 관측을 위한 주파수 합성기 설계·개발 등

2. 연구 내용

본 연구의 주요 내용은 다음과 같다.

□ e-VLBI 구현을 위한 DBBC¹⁾ 기능 고도화 및 최적화

- e-VLBI 관측을 위한 DBBC3 기능 고도화²⁾
- DBBC3의 측지관측 적용을 위한 VLBI 관측 소프트웨어 기능 개선
- DDC_U모드 확장에 따른 ‘VLBI 관측 프로그램 등’ 상호 연동을 위한 관련 소프트웨어 업데이트
- DBBC3 기반 VLBI 관측 안정화를 위한 성능 평가 및 최적화 수행

□ DiFX 상관처리 자동화 기술 고도화

- 상관처리를 위한 표준전파원 DB 구축
- 표준전파원 자동 선택·등록 유틸리티 개발
- 상관처리를 위한 지연시간 보정 기능 자동화
- VLBI 관측 데이터의 저장·관리·처리 자동화를 위한 관련 유틸리티 기능 업그레이드

□ e-VLBI 관측 안정화를 위한 성능 평가

- DiFX 상관기 성능 개선을 위한 서버 추가* 연동, DiFX 상관기 SW 업데이트 등 관측 환경 조성
- DiFX 서버 증설을 통한 2Gbps(기존 1Gbps)으로 대역폭을 향상하여 해외망원경 추가나 VLBI 관측 데이터 속도 향상
- e-VLBI 성능 업그레이드 DiFX 상관기를 이용한 e-VLBI 상관처리 테스트 수행
- e-VLBI 운용을 위한 사용 매뉴얼 제작(사용자 교육 포함) 및 e-VLBI 활용 방안 제시

□ 세종 VLBI 활용 확대를 위한 국·내외 VLBI 네트워크 협력 기반 구축

- KVN·EAVN과 공동 관측·협력 방안(협력 사항 등) 마련
- 국·내외 VLBI 관측망 참여를 위한 세종 VLBI 시스템 성능 측정 및 정량화된 성능지표 생성

1) Digital Base Band Converter : 아날로그 방식의 신호 변환 및 샘플링 기능을 디지털화 하는 장비

2) 기존 DDC_L모드에서 U모드로 확장하여 국·내외 관측소와 고주파수 관측, 입력 신호 및 처리 확장 등 다양한 관측 모드 기능 추가

- 세종 + KVN · EAVN 등 VLBI 관측망을 이용한 활용 분야(측지, 천문, 우주개발 등) 발굴 · 제시

□ 광대역 e-VLBI 관측을 위한 주파수 합성기 개발 및 적용 · 운영방안 마련

- 세종 시스템에 적합한 합성기 개발을 위한 시스템 구성도 분석 · 평가
- 주파수 합성기 개발을 위한 요구 성능 설계(입 · 출력 신호, 주파수 별 위상안정도 등)
- 주파수 합성기 회로 설계 및 제작
- 주파수 합성기 종합성능(요구성능 적합성, 위상 안정도 등 평가) 테스트
- 세종 VLBI 시스템 적용을 위한 현장 성능 시험
- 22/43기가 광대역화 구현을 위한 주파수 합성기 적용 방안 마련

제2장

e-VLBI 구현을 위한 DBBC 기능 고도화 및 최적화

-
1. e-VLBI 관측을 위한 DBBC3 기능 고도화
 2. DBBC3의 측지관측 적용을 위한 VLBI 관측 소프트웨어 기능 개선
 3. DBBC3 기반 VLBI 관측 안정화를 위한 성능 평가 및 최적화 수행

1. e-VLBI 관측을 위한 DBBC3 기능 고도화

가. DBBC3-4L4H 업그레이드

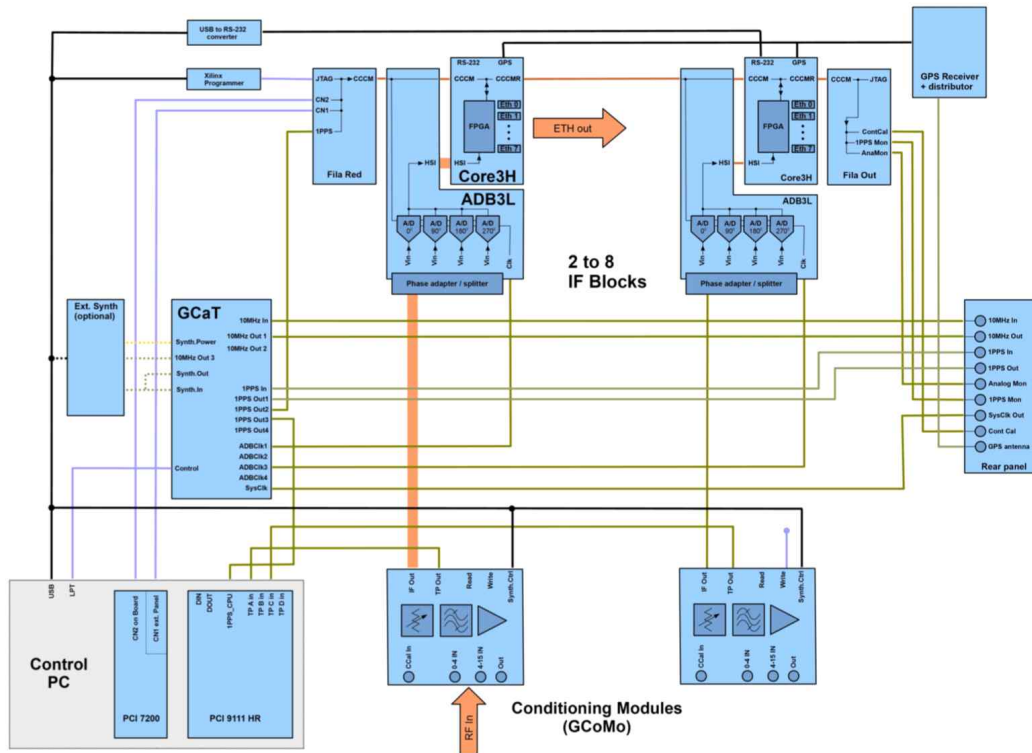


< 그림 2-1 > 세종 우주측지관측센터 DBBC3

우주측지관측센터에서는 관측 주파수 대역별로 2, 8, 22, 43GHz 대역의 4개의 수신기를 운영하고 있다. 국제 저주파 측지 관측용으로 2/8GHz를, 고주파 측지/천문 관측을 위해 22/43GHz 대역을 사용하고 있다. 22, 43GHz의 양편파를 동시에 사용할 수 있도록 2개의 Conditioning Module, 2 개의 샘플러 보드, 2개의 Core3H 보드를 추가하여 4개의 IF를 동시에 관측할 수 있도록 업그레이드하였다.

DBBC3는 Conditioning Module (GCoMo), ADB3L 샘플러 보드, Core3H 자료 처리 보드로 구성되어 있다. 하나의 ADB3L 보드는 4GHz 대역폭의 샘플링이 가능하다. ADB3L 출력 신호는 128Mbit/s로 Core3H 보드로 전달된다. 이번에 업그레이드를 통해 이러한 GCoMo, ADB3L, Core3H가 모두 2개에서 4개로 확장되었다.

DBBC3를 4L4H로 확장하였기 때문에 4개의 IF 처리가 가능해짐에 따라 4개의 IF를 동시에 사용하는 22GHz와 43GHz 전파 신호의 두 편광 성분인 좌편파(LCP)와 우편파(RCP)를 동시에 관측이 가능해졌으며, 향후 86GHz 수신기 도입시 22GHz, 43GHz, 86GHz의 단일 편파를 이용한 광대역 측지 관측이 가능해졌다.

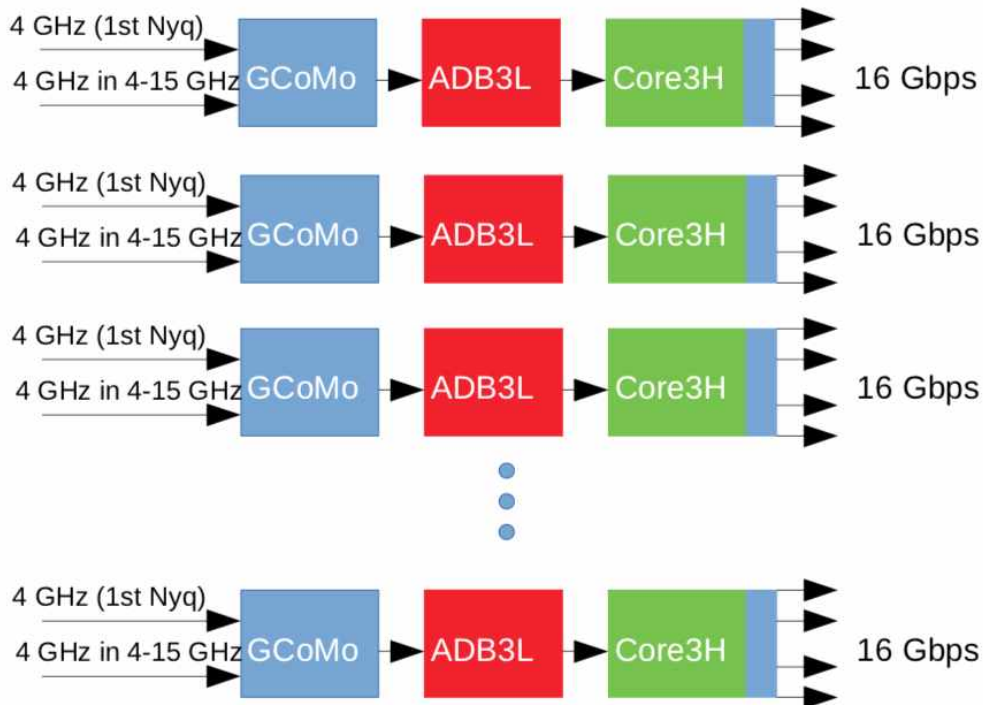


<그림 2-2> DBBC3 구성도 (DBBC3-4L4H는 4개의 IF Block 포함)

기존 세종의 DBBC3는 성능을 최대로 활용할 수 있는 DDC_U 모드와 OCT 모드의 경우 샘플링 주파수가 높아져서 많은 전력 소모와 열이 발생하여 안정적인 동작을 하지 못하였다. 이번 DBBC3를 업그레이드를 수행하면서 제작사에서 파워 모듈 등 시스템 전반에 대한 점검과 성능 테스트를 수행했으며, 따라서 그동안 안정적으로 사용할 수 없었던 DDC_U 모드와 OCT 모드를 사용할 수 있게 되었다. 다음 표는 이들 모드에서 지원되는 대역폭과 채널 개수를 보여준다. 업그레이드된 DBBC3의 성능 테스트 결과는 3절에 정리하였다.

<표 1-1> DBBC3 firmware 별 지원 기능

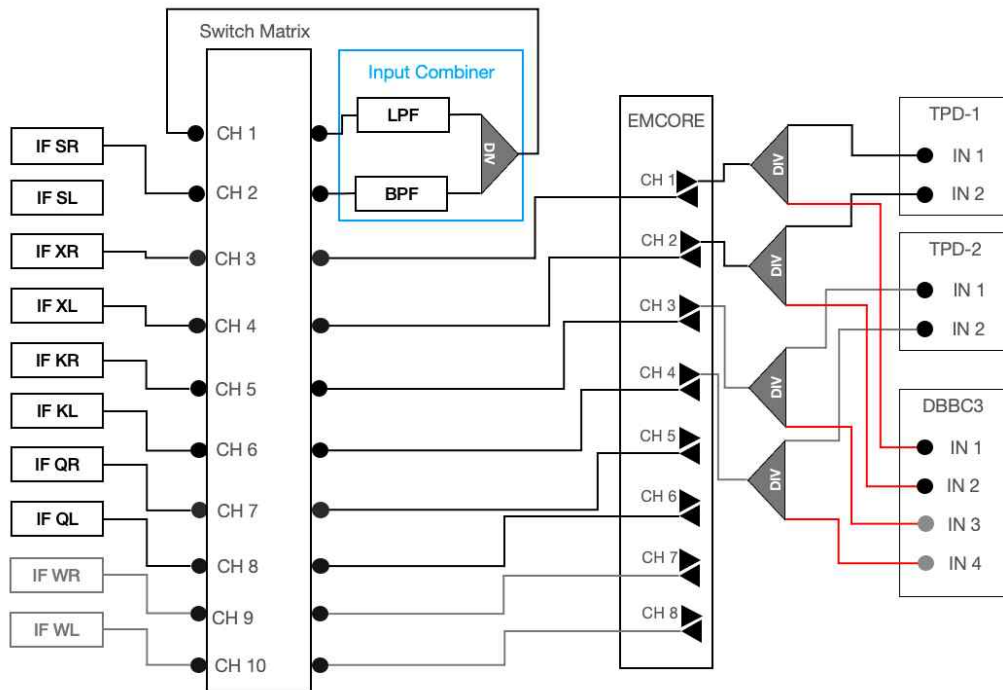
Firmware 별 지원 성능	
DDC_U (Unified)	<ul style="list-style-type: none"> BW : 1,2,4,8,16,32,128MHz Up to 16 BBCs/IF
OCT_D	<ul style="list-style-type: none"> Double 32-tap FIR-Filter/IF 512MHz BW : 0-512, 512-1024, .. 3584-4096 1024MHz BW : 0-1024, .. ,3072-4096 2048MHz BW : 0-2048, 2048-4096



<그림 2-3> DBBC3 입출력 구성

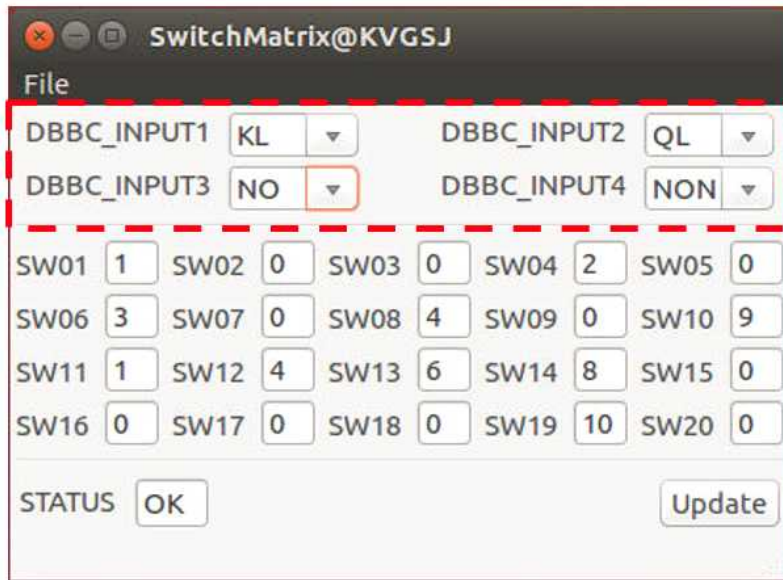
DBBC3는 위 그림과 같이 0-4GHz 대역 신호 또는 4-15GHz 대역 2개의 입력을 지원한다. 세종 우주측지관측센터는 수신기 방에서 IF 주파수를 낮은 Baseband 신호 (< 2GHz)로 Down Conversion 한 뒤 광신호 전송 장치를 통해 관측기기실로 전송한다. 관측기기실에서 광신호는 다시 0-2GHz 대역의 Baseband 신호로 변환하여 DBBC3의 0-4GHz 대역 입력 포트에 연결된다. 세종 DBBC3의 업그레이드를 수행하면서 최대 4GHz 입력 범위를 지원하는 DDC_U 모드와 OCT 모드를 안정적으로 사용할 수 있게 되었다. 하지만 세종 우주측지관측센터의 Down Converter의 출력 주파수 대역이 0-1GHz 대역 정도이고 광전송 모듈의 대역폭은 2.5GHz 이하로 제한되어있다. 따라서 DBBC3의 4GHz 대역폭을 최대한 활용하기 위해서는 Down Converter의 필터와 증폭기를 4GHz 대역폭을 가지는 것으로 교체해야 한다.

나. DBBC3-4L4H 확장에 따른 입력 연결 설정 소프트웨어 개선



<그림 2-4> 입력 결합 회로 구성안

위 그림과 같이 세종 우주측지 시스템은 현재 S, X, K, Q 밴드 수신기를 활용하고 있으며 각 수신기는 2개씩의 원형 편파 신호가 나온다. DBBC3-4L4H로 4개의 신호를 선택하기 위해 10개의 입력과 출력을 갖는 스위치 매트릭스가 있으며 이 스위치의 출력 신호는 광전송 모듈을 통해 관측기기실의 DBBC3로 입력된다. 세종 관측 소프트웨어에 10개의 IF 신호 중 4개의 신호를 선택하는 스위치 매트릭스 제어 모듈을 수정하였다. 특히 기존 소프트웨어에서는 안정적인 스위치 제어가 안되었는데, 기존 소프트웨어의 버그를 수정하여 신뢰성 있는 제어가 가능하게 되었다. 다음은 개발한 스위치 매트릭스의 제어 GUI 창을 보여준다.



<그림 2-5> DBBC3 입력 선택을 위한 스위치 매트릭스 제어 창

VLBI 관측 소프트웨어는 vex 파일을 해석하여 관측에 필요한 주파수의 IF 신호를 선택하도록 스위치 매트릭스를 제어하여 DBBC3 입력으로 설정한다. 관측 스크립트는 기존 측지 관측에 필요한 S/X 대역과 K/Q 대역 관측도 모두 자동 설정을 지원한다. DBBC3 4개의 보드로 확장되면 수신기 출력과 입력 신호 연결을 그림과 같이, 관측 가능 주파수 대역을 선택할 수 있는 스위치 매트릭스를 통해 연결함으로써 S/X 측지 관측뿐 아니라 K/Q 양편파(전자기파의 좌편파와 우편파를 통칭) 관측까지 원활한 지원이 가능해졌다.

다. e-VLBI 관측에 필요한 출력 설정 정의 및 소프트웨어 인터페이스 추가

e-VLBI 관측을 원활하게 수행하기 위해서는 DBBC3에서 출력되는 VDIF 파일이 지정된 상관기의 IP/Port로 출력되어야 한다. 즉 DBBC3의 출력을 손쉽게 설정할 수 있도록 e-VLBI 출력 설정을 정의하고 이를 제어할 인터페이스를 구현하였다. 아래 그림은 DBBC3의 출력을 제어하는 Core3H 제어 패널의 GUI를 보여준다. 4개의 Core3H에 대한 출력 제어를 각각 할 수 있으며 출력 IP/Port를 각각 설정할 수 있다. 허용된 IP와 Port는 미리 등록된 정보로부터 선택하도록 되어있어 간단하게 설정이 가능하다. 허용된 IP와 Port는 Mark6-1과 Mark6-2의 10GbE 입력 2개씩과 GPU 분광기 서버의 10GbE 입력이 등록되어있으며 e-VLBI를 위한 상관처리 서버도 등록되어있다. 해당 서버의 IP와 MAC Address를 선택하면 10GbE 출력이 상관처리 서버로 출력된다. e-VLBI는 네트워크 설정에 따라 Mark6를 경유해서 상관처리 서버로 전송하기도 하는데 이 경우는 Mark6로 출력을 설정후 Mark6에서 다시 상관 서버로 별도로 Forwarding 설정을 해줘야 한다.

The screenshot displays the Core3H control GUI with the following sections:

- Sampling Mode:** Mode (ddc_u), Sampling Rate (Mps) (32), Sampling Bit (2). [APPLY]
- PPS Delay:** ADB3L #1 #2 #3 #4 DELAY(ns) 32 32 32 32 [SYNC]
- DBBC IF:** Config button.
- DBBC IF Status:**

IF	Syn	Att	Mode	Target	Pwr
A	OFF	10	AGC	32000	32000
B	OFF	10	AGC	32000	32000
C	OFF	10	AGC	32000	32000
D	OFF	10	AGC	32000	32000
- DBBC IF Control:**

IF	Syn	Att	Target
A	off	agc	32000 SET
B	off	agc	32000 SET
C	off	agc	32000 SET
D	off	agc	32000 SET
- TIME / SYNC:** Date Time (22-06-01T 12:00:00+09:00), 1PPS Count (263), [SYNCD], sys, Sync.
- VDIFTX:** [STOP], Start, Stop.
- VSI Input:** Input Select (vsi1), Sample Rate (sps) (32000000), Decimation (1), Bitmask1 (0x000ffff), Bitmask2 (0x000ffff), Bitmask3 (0x000ffff), Bitmask4 (0x000ffff), SplitMode (off).
- VDIF Frame:** BitWidth (1), Num of Channels (1), Payload Size (8192), Corner Turning (off).
- XG Destination:**

Index	IP Address	
#1	0	10.10.1.10:46227 Set
#2	0	10.10.1.10:46227 Set
#3	0	10.10.1.10:46227 Set
#4	0	10.10.1.10:46227 Set
- XG Arp Config:**

Dev	Index	MAC Address	
#1	eth0	10	00:60:DD:43:1F:A8 Set
#2	eth1	26	00:60:DD:43:1F:A9 Set
#3	eth2		Set
#4	eth3		Set
- XG Config:**

Dev	IP	Gateway	Netmask	
#1	eth0	10.10.1.5	10.10.1.1	34 Set
#2	eth1	10.10.1.5	10.10.1.1	34 Set
#3	eth2	10.10.1.5	10.10.1.1	34 Set
#4	eth3	10.10.1.5	10.10.1.1	34 Set

<그림 2-6> Core3H의 출력 IP/Port 설정 제어 창

H_1	Core3H_2	Core3H_3	Core3H_4	SmpBit																														
<div>XG Destination</div> <table border="1"> <thead> <tr> <th></th> <th>Index</th> <th>IP Address</th> <th></th> </tr> </thead> <tbody> <tr> <td># 1</td> <td>0</td> <td>10.10.1.10:46227</td> <td>Set</td> </tr> <tr> <td># 2</td> <td>0</td> <td>10.10.1.10:46227</td> <td>Set</td> </tr> <tr> <td># 3</td> <td>0</td> <td>10.10.1.26:46228</td> <td>Set</td> </tr> <tr> <td># 4</td> <td>0</td> <td>10.10.1.27:46227</td> <td>Set</td> </tr> </tbody> </table>						Index	IP Address		# 1	0	10.10.1.10:46227	Set	# 2	0	10.10.1.10:46227	Set	# 3	0	10.10.1.26:46228	Set	# 4	0	10.10.1.27:46227	Set										
	Index	IP Address																																
# 1	0	10.10.1.10:46227	Set																															
# 2	0	10.10.1.10:46227	Set																															
# 3	0	10.10.1.26:46228	Set																															
# 4	0	10.10.1.27:46227	Set																															
<div>XG Arp Config</div> <table border="1"> <thead> <tr> <th></th> <th>Dev</th> <th>IP Address</th> <th></th> </tr> </thead> <tbody> <tr> <td># 1</td> <td>eth0</td> <td>10.10.1.41:46231</td> <td>Set</td> </tr> <tr> <td># 2</td> <td>eth1</td> <td>210.110.232.5:12003</td> <td>Set</td> </tr> <tr> <td># 3</td> <td>eth2</td> <td></td> <td>Set</td> </tr> <tr> <td># 4</td> <td>eth3</td> <td></td> <td>Set</td> </tr> </tbody> </table>						Dev	IP Address		# 1	eth0	10.10.1.41:46231	Set	# 2	eth1	210.110.232.5:12003	Set	# 3	eth2		Set	# 4	eth3		Set										
	Dev	IP Address																																
# 1	eth0	10.10.1.41:46231	Set																															
# 2	eth1	210.110.232.5:12003	Set																															
# 3	eth2		Set																															
# 4	eth3		Set																															
<div>XG Config</div> <table border="1"> <thead> <tr> <th></th> <th>Dev</th> <th>IP</th> <th>Gateway</th> <th>Netmask</th> <th></th> </tr> </thead> <tbody> <tr> <td># 1</td> <td>eth0</td> <td>10.10.1.5</td> <td>10.10.1.1</td> <td>34</td> <td>Set</td> </tr> <tr> <td># 2</td> <td>eth1</td> <td>10.10.1.5</td> <td>10.10.1.1</td> <td>34</td> <td>Set</td> </tr> <tr> <td># 3</td> <td>eth2</td> <td>10.10.1.5</td> <td>10.10.1.1</td> <td>34</td> <td>Set</td> </tr> <tr> <td># 4</td> <td>eth3</td> <td>10.10.1.5</td> <td>10.10.1.1</td> <td>34</td> <td>Set</td> </tr> </tbody> </table>						Dev	IP	Gateway	Netmask		# 1	eth0	10.10.1.5	10.10.1.1	34	Set	# 2	eth1	10.10.1.5	10.10.1.1	34	Set	# 3	eth2	10.10.1.5	10.10.1.1	34	Set	# 4	eth3	10.10.1.5	10.10.1.1	34	Set
	Dev	IP	Gateway	Netmask																														
# 1	eth0	10.10.1.5	10.10.1.1	34	Set																													
# 2	eth1	10.10.1.5	10.10.1.1	34	Set																													
# 3	eth2	10.10.1.5	10.10.1.1	34	Set																													
# 4	eth3	10.10.1.5	10.10.1.1	34	Set																													

<그림 2-7> DBBC3의 Core3H 10GbE 출력 설정

2. DBBC3의 측지 관측 적용을 위한 VLBI 관측 소프트웨어 기능 개선

가. 측지 관측 적용을 위한 DBBC3 인터페이스 확장 및 개선

The screenshot displays the DBBC3 GUI with several configuration panels:

- Sampling Mode:** Includes a dropdown for Mode (set to 'dbs_u'), a dropdown for Sampling Rate (Msps) (set to 32), and a dropdown for Sampling Bit (set to 2). An 'APPLY' button is present.
- PPS Delay:** A table for ADB3L delays across four channels (#1, #2, #3, #4) with a 'SYNC' button.
- DBBC IF:** A 'Config' button and a 'DBBC IF Status' table.
- DBBC IF Control:** A table for controlling IF parameters (IF, Syn, Att, Mode, Target, Pwr) for channels A, B, C, and D, with an 'Update' button.
- DBBC Control:** A large table for configuring DBBC channels (CH1 to CH72) with columns for FREQ, IF, BW, TPint, and SET.
- DBBC Gain Control:** A table for gain control (Mode/GainU, GainL, 0-255) for channels CH1 to CH72, with 'Set' buttons.
- DBBC Gain Status:** A table for monitoring gain status (Mode, GainU, GainL, TPUON, TPLON, TPUOFF, TPLOFF).

<그림 2-8> DBBC3의 DBBC 주파수 채널 설정 GUI 창

DBBC3의 4L4H로 확장됨에 따라 기존에 DBBC 신호처리 Board와 Core3H VDIF 출력 보드를 각각 2개씩 제어하던 기능을 4개씩 확장하였다. 또한 DBBC의 IF 입력 제어 및 상태 창도 4개의 IF에 대해 제어 및 상태 표시가 가능하도록 확장하였다.

File

Sampling Mode
Mode: ddc_u
Sampling Rate (MSPS): 32
Sampling Bit: 2
APPLY

PPS Delay
ADB3L #1 #2 #3 #4
DELAY(ns) 32 32 32 32 SYNC

DBBC IF Config
DBBC IF Status
IF Syn Att Mode Target Pwr
A OFF 10 AGC 32000 32000
B OFF 10 AGC 32000 32000
C OFF 10 AGC 32000 32000
D OFF 10 AGC 32000 32000

DBBC IF Control
IF Syn Att Target
A off agc 32000 SET
B off agc 32000 SET
C off agc 32000 SET
D off agc 32000 SET
Update

TIME / SYNC
Date Time: 22-06-01T 12:00:00+09:00
1PPS Count: 263
SYNCED sys Sync

VDIF TX
STOP Start Stop

VSI Input
Input Select: vsi1
Sample Rate (SPS): 32000000
Decimation: 1
Bitmask1: 0x000fffff
Bitmask2: 0x000fffff
Bitmask3: 0x000fffff
Bitmask4: 0x000fffff
Split Mode: off

VDIF Frame
Bit Width: 1
Num of Channels: 1
Payload Size: 8192
Corner Turning: off

XG Destination
Index IP Address
#1 0 10.10.1.10:46227 Set
#2 0 10.10.1.10:46227 Set
#3 0 10.10.1.10:46227 Set
#4 0 10.10.1.10:46227 Set

XG Arp Config
Dev Index MAC Address
#1 eth0 10 00:60:DD:43:1FA8 Set
#2 eth1 26 00:60:DD:43:1FA9 Set
#3 eth2 Set Set
#4 eth3 Set Set

XG Config
Dev IP Gateway Netmask
#1 eth0 10.10.1.5 10.10.1.1 34 Set
#2 eth1 10.10.1.5 10.10.1.1 34 Set
#3 eth2 10.10.1.5 10.10.1.1 34 Set
#4 eth3 10.10.1.5 10.10.1.1 34 Set
Update

Update Set

<그림 2-9> Core3H의 출력 IP/Port 설정 제어 창

DDC_U Firmware는 하나의 ADC 보드에 있는 4개의 샘플러에 자료를 합쳐 광대역 신호를 만드는데 이 4개의 샘플러 보드에 있는 4개 샘플러의 입력 신호의 세기와 동기화 상태를 한번에 확인할 수 있도록 SmpBit 창에 4개의 Core3H에 입력되는 신호의 세기 분포를 표시하도록 확장하였다.

DBBC_1 DBBC_2 DBBC_3 DBBC_4 Core3H_1 Core3H_2 Core3H_3 Core3H_4 SmpBit

Core3H_1

Power
#0 250000
#1 250000
#2 250000
#3 250000

Bstat
00 01 10 11
#0 25 25 25 25
#1 25 25 25 25
#2 25 25 25 25
#3 25 25 25 25

Corr
0-1 15000
1-2 15000
2-3 15000

Core3H_2

Power
#0 250000
#1 250000
#2 250000
#3 250000

Bstat
00 01 10 11
#0 25 25 25 25
#1 25 25 25 25
#2 25 25 25 25
#3 25 25 25 25

Corr
0-1 15000
1-2 15000
2-3 15000

Core3H_3

Power
#0 250000
#1 250000
#2 250000
#3 250000

Bstat
00 01 10 11
#0 25 25 25 25
#1 25 25 25 25
#2 25 25 25 25
#3 25 25 25 25

Corr
0-1 15000
1-2 15000
2-3 15000

Core3H_4

Power
#0 250000
#1 250000
#2 250000
#3 250000

Bstat
00 01 10 11
#0 25 25 25 25
#1 25 25 25 25
#2 25 25 25 25
#3 25 25 25 25

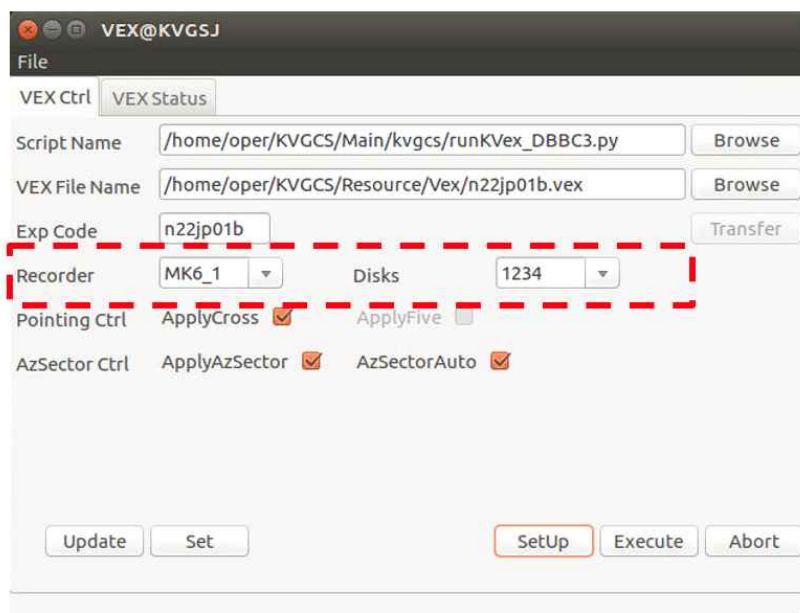
Corr
0-1 15000
1-2 15000
2-3 15000

<그림 2-10> Core3H_1,2,3,4 의 입력 신호 세기 분포 표시 창

나. DBBC3 업그레이드에 따른 관측 소프트웨어 기능 개선

1) 기록기와 디스크 모듈 설정 기능 개선

DBBC3 확장에 따라 여러 VDIF 출력 스트림에 대해 다양한 설정을 할 수 있어야 한다. DBBC3의 출력 VDIF 스트림은 Thread ID가 헤더에 포함되어 있지 않기 때문에 여러 Thread를 하나로 묶여있는 디스크 모듈에 기록할 경우 상관 처리 시 서로 섞여 있는 Thread를 구분하기 어려워진다. 이를 피하기 위해 Mark6에서 각 최대 4개의 Thread를 디스크 모듈에 각각 하나씩 나누어 기록할 수 있다. 두 개의 Thread를 4개의 디스크 모듈 묶음에 기록하도록 할 경우 각각의 thread를 디스크 모듈 12와 34에 하나씩 나누어 기록하도록 VLBI 관측 소프트웨어에 디스크 모듈 분할 기능을 추가하였다. Data Rate에 따라 사용할 디스크 모듈도 선택할 수 있도록 기능을 개선하였다.



<그림 2-11> VLBI 관측 스크립트 실행 창

Mark6는 4개의 디스크 모듈을 가질 수 있으므로 4개의 독립적인 입력 스트림을 개별 디스크 모듈에 기록할 수 있다. 4개의 입력 스트림을 제어하고 상태를 표시할 수 있도록 소프트웨어에 기능을 추가하였다. runKVex_DBBC3.py 스크립트는 선택된 기록기와 디스크 모듈에 따라 자동으로 DBBC3의 출력 IP/Port를 설정한다.

아래 그림은 2개의 입력 스트림을 12와 34의 두 개 그룹으로 나누어 기록하는 Mark6 상태를 보여준다. Mark6-1에 해당되는 IP/Port로 입력 스트림이 설정되었으며 Thread 개수에 맞게 디스크 모듈도 분할되어 선택된 것을 보여준다.

The screenshot shows the Mark6 software interface. At the top, there's a title bar 'MK6@KVGJSJ' and a 'File' menu. Below it are tabs for 'MK6', 'MK6_1', and 'MK6_2'. The main area is divided into several sections:

- MStat:** A table showing disk status for four slots.

Slot	Grp	eMSN	Size(GB)	Status	Protect	Type
1	1234	KVG00003/32000/4/8	28472	open	ready	sg
2	1234	KVG00004/32000/4/8	28472	open	ready	sg
3	1234	KVG00001/32000/4/8	28474	open	ready	sg
4	1234	KVG00002/32000/4/8	28474	open	ready	sg
- Input Stream:** A table showing input stream details.

Src	Format	Payload	Interface	IPAddress	Disk
core3h_	vdif	8224/50	eth3	10.10.1.21/46228	34
core3h_	vdif	8224/50	eth2	10.10.1.5/46227	12
- Remaining HDD Space and Time:** A table showing disk space and time metrics.

Time(min)	Space(GB)	Percent(%)	Rate(Mbps)
14772.5	113896	89.0	1028.0
- Last Scan:** A table showing the last scan results.

Label	1234	3052	a2210f_KVGJSJ_10519170f	vdif	58.9640
Status	OK	458.866	2022y105d19h17m23s	1.02800	0

<그림 2-12> Mark6 상태 창 - 4개 입력 스트림 정보 표시

위와 같이 2개의 디스크 모듈로 묶어 기록된 Mark6의 Scatter 파일은 나중에 fuse로 Linux 파일로 만들 때 기록된 디스크 모듈 별로 마운트를 해서 읽어야 한다. 다음은 Mark6에서 디스크 모듈 그룹별로 fuse 마운트/언마운트를 위한 축약 명령어 설정을 보여준다.

디스크 모듈 1234

```
alias fusemk6='fuseMk6 -r "/mnt/disks/[1-4]/[0-7]/data/" /mnt/mark6sg'
```

```
alias fusemk6u='fusermount -u /mnt/mark6sg'
```

디스크 모듈 12, 34

```
alias fusemk6_12='fuseMk6 -r "/mnt/disks/[12]/[0-7]/data/" /mnt/mark6sg_12'
```

```
alias fusemk6u_12='fusermount -u /mnt/mark6sg_12'
```

```
alias fusemk6_34='fuseMk6 -r "/mnt/disks/[34]/[0-7]/data/" /mnt/mark6sg_34'
```

```
alias fusemk6u_34='fusermount -u /mnt/mark6sg_34'
```

디스크 모듈 1, 2, 3, 4

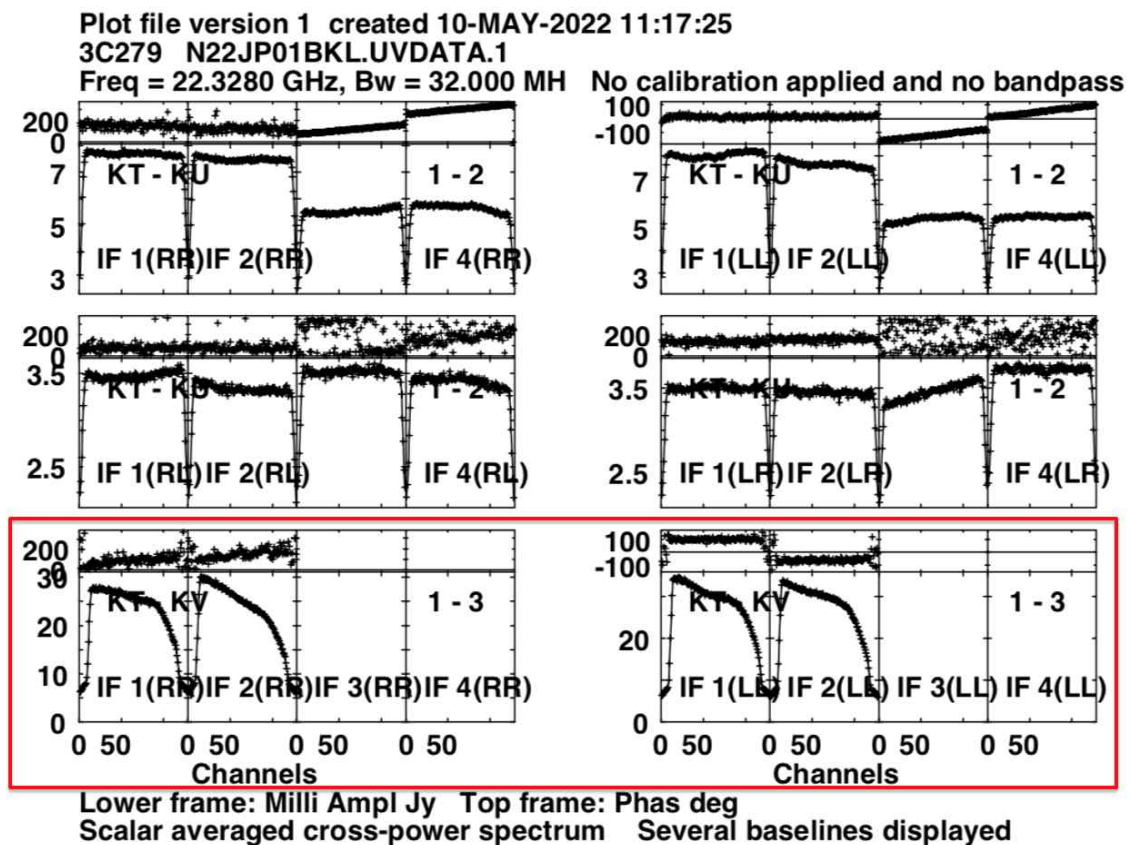
```
alias fusemk6_1='fuseMk6 -r "/mnt/disks/1/[0-7]/data/" /mnt/mark6sg_1'
```

```
alias fusemk6u_1='fusermount -u /mnt/mark6sg_1'
```

```
alias fusemk6_2='fuseMk6 -r "/mnt/disks/2/[0-7]/data/" /mnt/mark6sg_2'
```

```
alias fusemk6u_2='fusermount -u /mnt/mark6sg_2'
alias fusemk6_3='fuseMk6 -r "/mnt/disks/3/[0-7]/data/" /mnt/mark6sg_3'
alias fusemk6u_3='fusermount -u /mnt/mark6sg_3'
alias fusemk6_4='fuseMk6 -r "/mnt/disks/4/[0-7]/data/" /mnt/mark6sg_4'
alias fusemk6u_4='fusermount -u /mnt/mark6sg_4'
```

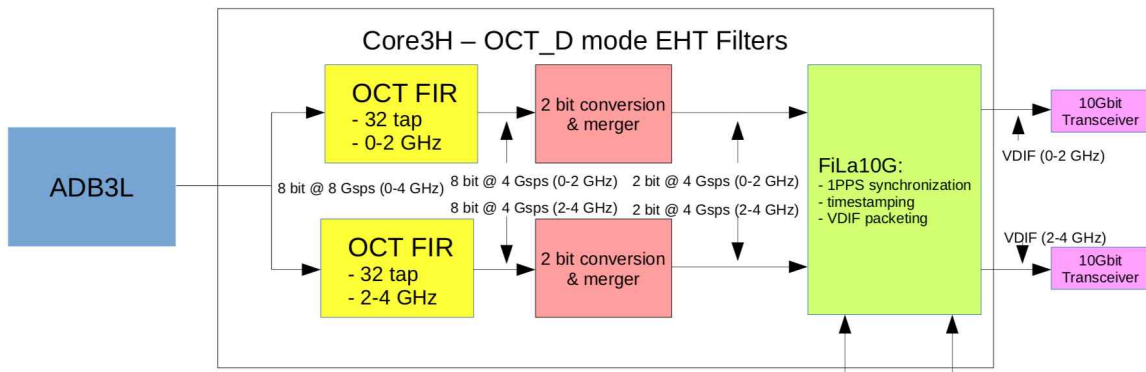
업그레이드된 관측 소프트웨어로 2개의 IF를 이용한 편광 관측 n22jp01b의 프린지 결과는 아래 그림과 같다. 이 관측은 아직 DBBC3 업그레이드 전에 수행한 관측으로 DDC_L firmware를 이용하였지만, 이 관측을 통해 업그레이드된 관측 소프트웨어의 기능을 검증할 수 있다. KVN 탐라(KT)와 우주측지관측센터(KV) 사이에서 LL과 RR에 대해 모두 뚜렷한 프린지를 검출하였음을 볼 수 있다. 이로써 업그레이드된 관측 소프트웨어에서 Switch Matrix 제어 기능, DBBC 주파수 설정 기능, 복수의 DBBC 출력 IP/Port 설정 기능, 기록기의 스트림 정보 및 디스크 모듈 제어 등 모든 기능에 대해 정상적으로 동작하는 것을 확인하였다.



<그림 2-13> 세종 망원경을 이용한 22GHz 양편파 관측에 대한 프린지 결과

2) 광대역 관측 지원 OCT 모드 인터페이스 추가

DDC_U firmware가 128MHz 이하의 대역폭을 갖는 모드를 지원하는 반면 OCT firmware는 512MHz, 1024MHz, 2048Hz 대역폭을 갖는 광대역 모드를 지원한다. DBBC3 업그레이드에 따라 기존에 사용할 수 없었던 광대역 관측 모드인 OCT 모드를 사용할 수 있게 되었다. OCT 모드에서는 아래 그림과 같이 각 보드 당 두 개의 필터의 형태를 설정하는 탭을 설정할 수 있는데 0-4000MHz 대역 범위 안에서 필요한 대역폭과 시작 주파수를 설정할 수 있도록 탭 파일이 DBBC3 제어 컴퓨터에 저장되어 있으며 Tap 명령과 Tap2 명령으로 2개의 필터 탭 파일을 선택할 수 있다.



<그림 2-14> OCT_D firmware의 동작 구성

DBBC_1	DBBC_2	DBBC_3	DBBC_4	Core3H_1	Core3H_2	Core3H_3	Core3H_4	SmpBit	OCT Tap
OCT TAP Control									
BOARD	BW (MHz)		FSTART (MHz)		SCALE				
#1	500	▼	0	▼	1.0	SET			
	500	▼	500	▼	1.0	SET			
#2	500	▼	0	▼	1.0	SET			
	500	▼	500	▼	1.0	SET			
#3	500	▼	0	▼	1.0	SET			
	500	▼	500	▼	1.0	SET			
#4	500	▼	0	▼	1.0	SET			
	500	▼	500	▼	1.0	SET			
Update									

<그림 2-15> OCT_D 모드의 필터 탭 설정 GUI 패널

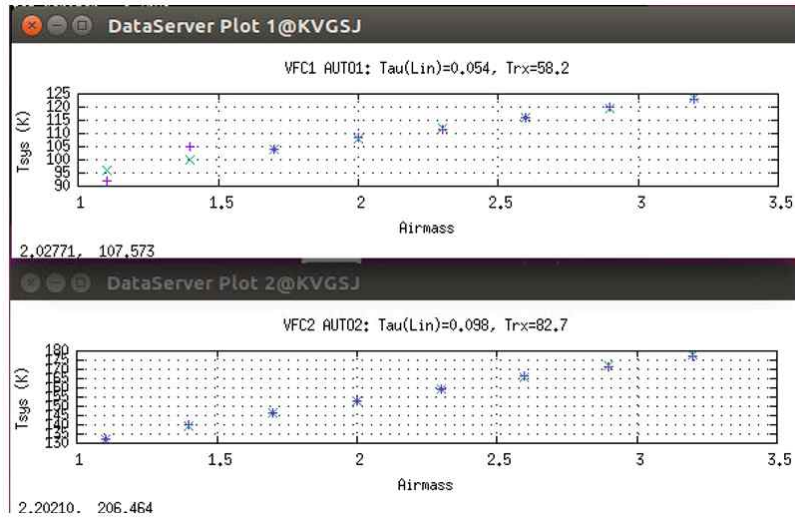
관측 소프트웨어 GUI에 OCT 모드의 탭 파일을 선택할 수 있도록 OCT Tap 패널을 추가하였으며 각 보드 당 2 개씩의 필터 탭을 선택할 수 있다. runKVex_DBBC3.py VLBI 관측 스크립트는 vex 파일을 해석하여 512MHz 이상의 광대역 관측 모드 일 때 필터 탭을 자동으로 설정한다.

3) Total Power Detector 기준으로 자료 처리 프로그램 (DSL0c) 개선

기존 자료 처리 서버인 DSL0c에서는 전파분광기 신호를 기준으로 모든 자료 처리의 흐름이 제어되었다. 즉 신호의 적분 기간, 적분 시작과 종료, 캘리브레이션 등이 모두 전파분광기 출력신호를 기준으로 처리가 됐다. 주로 분광 관측이 중요한 시스템에서는 이러한 처리 방식이 적절하지만 VLBI와 같이 관측 중 분광 자료가 필요가 없는 경우는 VLBI 자료의 신호 세기의 캘리브레이션에 사용되는 백엔드를 기준으로 자료 처리를 수행하는 게 더 안정성과 확장성을 높일 수 있다. DBBC3 업그레이드에 따라 4개의 광대역 출력 신호에 대한 처리가 필요한데 현재 세종에서 사용하는 GPU 분광기로는 DBBC3에서 출력하는 신호에 대한 실시간 분광처리는 제한될 뿐만 아니라, DBBC3 출력 신호의 기록 또는 상관처리를 위해 Mark6 기록기나 상관 서버로 VDIF 신호가 전송되는 경우 분광기는 입력 신호가 없어서 실제 VLBI 관측 기록 중에 신호에 대한 캘리브레이션을 할 수 없다.

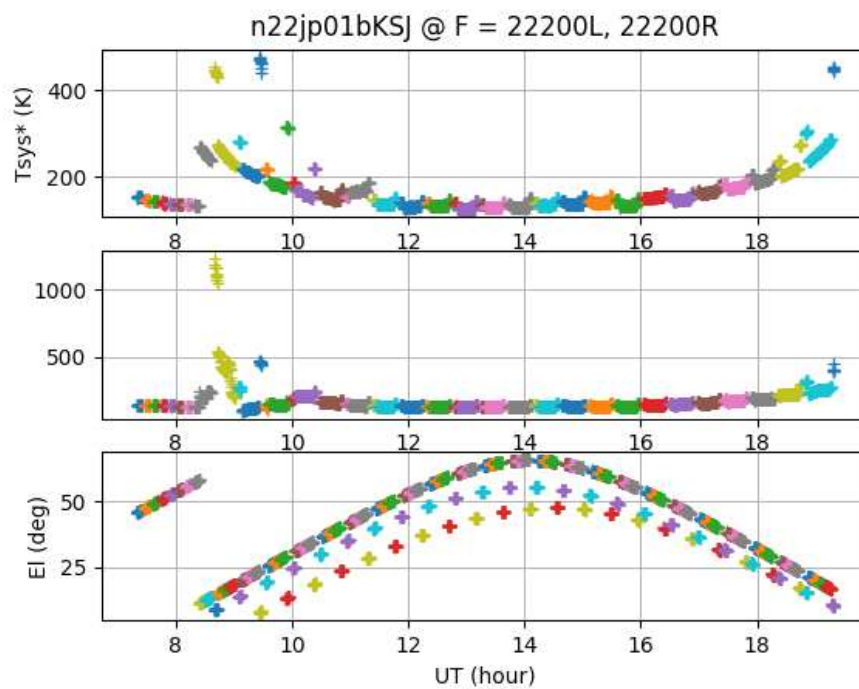
세종 우주측지관측센터에서는 VLBI 관측 자료 처리에 필요한 Amplitude 캘리브레이션 자료는 Total Power Detector의 값을 이용하고 있다. 따라서 DSL0c의 자료 처리 흐름의 기준을 모두 Total Power Detector 자료로 변경하였다. 세종 우주측지관측센터는 4개의 Total Power Detector가 설치되어있어서 동시에 4개 IF 채널에 대한 캘리브레이션이 가능하다.

또한 이번 변경을 통해 실시간으로 시스템 잡음온도(T_{sys})를 계산해서 저장하는 기능을 추가하였다. VLBI 스캔 중에 5초 간격으로 T_{sys} 값을 업데이트하고 이 값을 스캔 번호로 주어진 파일($\sim/KVGCS/Data/vfc/tsys_<scannum>.dat$) 이름으로 저장된다. 이 T_{sys} 자료 파일은 나중에 ANTAB 파일 양식과 동일하게 DOY, UT_TIME, TSYS1, TSYS2 순으로 저장되어 있어 ANTAB 파일 생성도 단순화되었다.



<그림 2-16> 자료 처리 서버에서 Total Power Detector (VFC) 값을 이용한 대기 흡수 계수 측정 결과

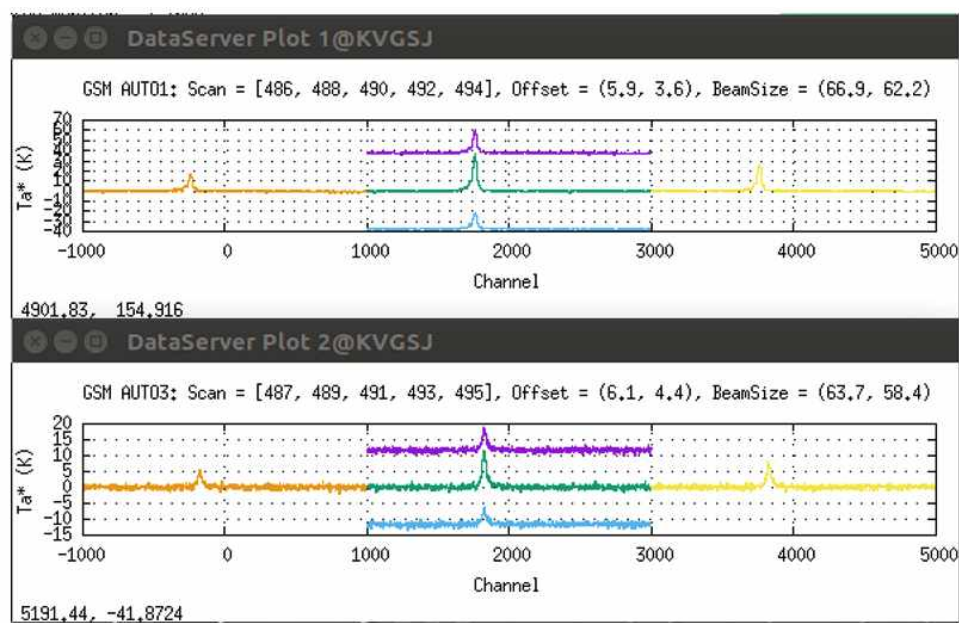
다음은 Tsys 자료 파일을 이용해서 생성한 ANTAB 파일의 내용을 Plot 이다. 22GHz 양편과 관측에 대해 2개의 IF에 대해 독립적으로 캘리브레이션이 수행되었다.



<그림 2-17> 업그레이드된 DSLoc 에서 기록한 시스템 온도 자료를 이용한 ANTAB 파일 결과

3) GSM 분광기 자료 처리

전과 관측을 통해 신뢰할 수 있는 자료를 얻기 위해서는 안테나의 지향 정밀도와 초점 측정 기능이 필수적이다. 기존 DSLoc 변경과 더불어 GPU 분광기의 자료 처리 과정도 지향 정밀도 관측과 초점 측정을 수행할 수 있도록 수정하였다. 다음은 업그레이드된 DSLoc에서 GPU 분광기 자료를 이용해서 얻은 지향 정밀도 측정 결과를 보여준다. 43GHz 대역에 있는 43.122GHz SiO J=1-0, v=1 메이저 방출선과 42.822GHz에 있는 SiO J=1-0, v=2 메이저 방출선을 동시에 얻을 수 있으며 각각에 대해 지향 오프셋을 측정해서 표시해주는 것을 볼 수 있다.



<그림 2-18> 업그레이드된 DSLoc를 이용한 Five Pointing 관측 결과 표시

3. DBBC3 기반 VLBI 관측 안정화를 위한 성능 평가 및 최적화 수행

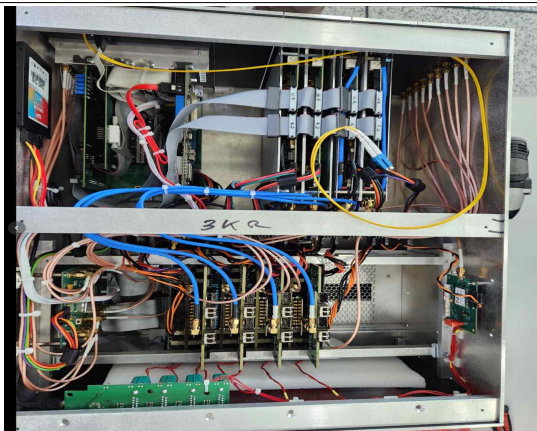
기존 세종 DBBC3는 DDC_L 모드의 firmware를 사용하여 관측을 수행해 왔다. DDC_L 모드는 최대 1GHz 대역폭을 처리할 수 있는데, 수신기의 LO 주파수와 DDC_L 모드의 제한된 대역폭으로 인해 관측 가능한 주파수 대역이 21.0~21.9GHz로 제한된다. 이러한 성능의 제약을 극복하고 VLBI 관측 안정화를 위하여, DBBC3 업그레이드 및 최적화를 수행하였다. DDC_U 모드와 OCT 모드를 지원할 경우, 샘플링 주파수가 높아져서 많은 전력 소모와 열이 발생하는데, 제작사(HAT-Lab) 및 독일 막스플랑크 전파연구소에서 파워 모듈과 케이블 등 시스템 전반에 대한 점검과 성능 테스트를 완료하였다. 이 성능 점검 결과 평가서는 부록 1에 수록하였다. 아래 그림은 업그레이드 된 DBBC3가 세종 우주측지관측센터의 기기실에 설치되어 작동하는 모습이다.



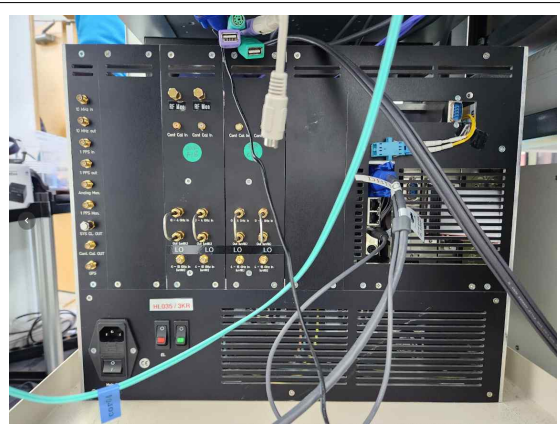
업그레이드 된 세종 DBBC3



기기실에 장착된 세종 DBBC3



기존 보드 2개에서 4개로 확장된 DBBC3 내부 구조 모습



기존 보드 2개에서 4개로 확장된 DBBC3 후면 패널 모습

<그림 2-19> 업그레이드된 DBBC3 모습 (세종 우주측지관측센터)

제3장

DiFX 상관처리 자동화 기술 고도화

VLBI 데이터 분석 가속화 기술개발 연구[2차]

-
1. 상관처리를 위한 표준전파원 DB 구축
 2. 표준전파원 자동 선택 · 등록 유틸리티 개발
 3. 상관처리를 위한 지연시간 보정 기능 자동화
 4. VLBI 관측 데이터의 저장·관리·처리 자동화를 위한 관련 유틸리티
기능 업그레이드

1. 상관계리를 위한 표준전파원 DB 구축

VLBI 관측은 관측 데이터의 세기를 표준화하기 위하여 관측 시작과 끝에 표준전파원(일명 캘리브레이터)을 관측함으로써 데이터 세기를 표준화하는 작업을 수행한다. 캘리브레이터는 관측 중간 중간에도 관측하는 것이 관측 데이터의 표준화에 훨씬 유용하므로 여러 표준전파원을 사용하기도 한다. 이러한 표준전파원은 미리 조사되어 여러 관측자에게 제공되고 있으며, 대부분의 VLBI 관측을 위한 vex 파일에는 이 표준전파원을 이용한 캘리브레이션 관측을 수행한다.

상관계리에서 clockrate 값을 정하기 위하여 표준전파원을 관측한 스캔을 이용하면 clockrate의 값을 안정적으로 결정할 수 있다. VLBI 관측에서 사용하는 관측 요청서인 vex 파일의 \$SCHD 블록을 분석하면 관측한 천체들을 알 수 있다. vex 파일을 분석하여 스캔별로 분류하고 이들 스캔들을 표준전파원 DB와 비교하면 표준전파원을 관측한 스캔들을 알 수 있다. 따라서, 표준전파원 DB를 구축하고, 이를 바탕으로 vex 파일에서 표준전파원 스캔들을 추출하여 처리함으로써 표준전파원을 이용한 clockrate 값을 결정하게 된다.

표준전파원 DB는 텍스트 기반의 평문으로 작성함으로써, 일반 텍스트 편집기로도 쉽게 편집할 수 있어야 한다. 일반적으로 표준전파원은 {천체, 좌표, 세기}의 값을 갖고 있다. 이러한 목록을 DB화함으로써, 상관계리시 clockrate 계산에 필요한 스캔 목록을 자동 생성할 수 있게 되고, 이를 기반으로 DiFX 상관계리의 자동화 기능이 확장하게 된다.

표준전파원 목록을 작성하기 위하여 여러 관측에서 사용되는 프린지 파인더를 참조하였다. 각 연구자마다 해당 연구에 필요한 표준전파원 목록을 선택하여 관측하기 때문에, 이러한 관측에서 사용하는 표준전파원 정보를 모아 표준전파원 목록을 작성하는 것이 가장 실용적이라 기대되기 때문이다.

표준전파원 목록은

```
이름; ra = ; dec = ; ref_coord_frame = J2000
```

와 같은 형식을 가지면, 일반 리눅스 텍스트 파일로 저장된다.

수집된 표준전파원 목록은 다음과 같다.

0055+ 300;	ra = 00h57m48.8833380s; dec = 30d21'08.811970"; ref_coord_frame = J2000;
0133+ 476;	ra = 01h36m58.5948050s; dec = 47d51'29.100020"; ref_coord_frame = J2000;
0235+ 164;	ra = 02h38m38.9301070s; dec = 16d36'59.274500"; ref_coord_frame = J2000;
0321+ 340;	ra = 03h24m41.1613190s; dec = 34d10'45.857140"; ref_coord_frame = J2000;
3C273;	ra = 12h29m06.6997295s; dec = 02d03'08.598285"; ref_coord_frame = J2000;
3C279;	ra = 12h56m11.1665682s; dec = -05d47'21.524998"; ref_coord_frame = J2000;
3C345;	ra = 16h42m58.8099630s; dec = 39d48'36.993850"; ref_coord_frame = J2000;
3C454.3;	ra = 22h53m57.7479370s; dec = 16d08'53.560930"; ref_coord_frame = J2000;
3C454;	ra = 22h53m57.7479370s; dec = 16d08'53.560930"; ref_coord_frame = J2000;
3C84;	ra = 03h19m48.1601140s; dec = 41d30'42.105580"; ref_coord_frame = J2000;
BLLAC;	ra = 22h02m43.2913710s; dec = 42d16'39.979870"; ref_coord_frame = J2000;
CTA102;	ra = 22h32m36.4089050s; dec = 11d43'50.903940"; ref_coord_frame = J2000;
M87;	ra = 12h30m49.4233826s; dec = 12d23'28.043742"; ref_coord_frame = J2000;
NRAO150;	ra = 03h59m29.7472658s; dec = 50d57'50.161621"; ref_coord_frame = J2000;
NRAO530;	ra = 17h33m02.7057870s; dec = -13d04'49.548400"; ref_coord_frame = J2000;
OJ287;	ra = 08h54m48.8749300s; dec = 20d06'30.640780"; ref_coord_frame = J2000;

새로운 표준전파원 목록은 이 텍스트파일 ‘calibrator.doc’에 추가되며, 표준전파원을 검색하는 파이썬 유틸리티에 의해 참조된다.

2. 표준전파원 자동 선택 · 등록 유틸리티 개발

관측자가 요구한 관측을 수행하기 위하여 관측 정보와 관측 스케줄이 포함된 vex 파일이 관측 프로그램에 제공된다. DIFX 상관처리에서는 이 vex 파일을 이용하여 상관처리를 진행시키며, 해당 \$SCHED; 블록에 나열된 천체 목록을 순서대로 상관처리하여 그 결과를 저장한다. 관측 \$SCHED; 블록에는 대개 표준전파원을 적당히 분포시켜 전파 세기를 표준화시키고 있으며, 이러한 표준전파원 관측을 이용하여 원하는 관측 천체의 세기를 표준화하여 연구에 활용한다. 즉, 관측을 수행하는 vex 파일의 \$SCHED; 블록에는 표준전파원이 포함되어 있다는 것이다. 따라서 이 \$SCHED; 블록을 분석하여 표준전파원에 해당되는 천체를 이용하여 프린지 검출을 시도하면 된다.

‘vexsched.py’ 파이썬 유틸리티를 이용하면 해당 표준전파원을 vex 파일에서 추출하여 프린지 검출을 위한 v2d 파일을 수정할 수 있도록 하였다.

```
vexsched.py -s "BLLAC 3C373"
```

와 같이 표준전파원을 이중괄호(“)로 묶어서 입력하면, vex 파일의 \$SCHED; 블록에서 관측 ‘source’가 ‘BLLAC’와 ‘3C273’인 관측 스캔들을 모두 숨아 내어 목록으로 출력시킨다. 이 출력 내용을 저장하고, 출력된 목록을 v2d 파일의 전체 목록에 추가하거나 대체하면 원하는 전체인 ‘BLLAC’와 ‘3C273’만을 처리하는 v2d 파일이 만들어지는 것이다.

3. 상관처리를 위한 지연시간 보정 기능 자동화

1차년도에서 개발된 N-포인트 방법을 이용한 clockrate 계산 과정에 표준전파원을 이용한 자동화 방식을 도입함으로써 상관처리의 지연시간을 보다 안정적으로 계산할 수 있게 될 것이다.

v2d 파일을 이용한 상관처리는 run.EXP.CORRbySCAN 스크립트를 실행시키면 v2d 파일에 등록된 관측스캔들을 차례로 상관처리한다. 상관처리가 완료되면, run.PARSE.apd_delays.sh 스크립트를 실행하여 각 스캔별로 계산된 지연시간을 수집하여 지연시간 데이터 파일로 정리하고, 해당 기선별 rate로 분류한다. ‘gnuplot gnuplotFS.dem’을 실행하면, 지연시간을 그림으로 출력해 주며, 육안으로 확인할 수 있다. run.clockrate.sh를 실행하면 분류된 지연시간을 이용하여 기선별 clockrate를 계산하게 되며, 지연시간을 제로화(최소화)하기 위한 지연시간 오프셋을 계산한다. 계산된 clockrate와 지연시간 오프셋을 update.clockrate.sh 스크립트에 의해 v2d 파일에 자동 업데이트(등록)된다.

최종 등록된 clockrate와 지연시간 오프셋을 이용하여 관측 전부를 처리하는 v2d 파일을 생성한다. 관측목록은 vex 파일의 관측 스캔 목록을 그대로 사용하면 될 것이다.

이러한 과정을 요약하면

- 표준전파원을 이용한 v2d 파일 작성
 - clockrate 및 지연시간 계산 및 결과 업데이트
 - 전체 처리를 위한 v2d 파일 작성 및 상관처리
- 으로 구분된다.

가. N-포인트 방법 자동화

프린지 검출을 수행하는 ‘run.EXP.CORRbySCAN’는 v2d 파일에 등록되어 있는 표준전파원을 상관처리하도록 자동화되어 있는 스크립트이다.

다음은 ‘run.EXP.CORRbySCAN’ 스크립트의 ‘help’ 내용을 보여주고 있다.

```
[oper@coma FS]$ ./run.EXP.CORRbySCAN -h
cmd>-h scmd="-h"
./run.EXP.CORRbySCAN <obscode>
option:
-O --obscode <obscode>
-M --mode each
-S --scan : enable scanfmt_if_possible (default FALSE)
--reverse : sort by reverse order
--every : sort by every other
--pingpong : sort by center : 1,n,2,n-1,...
--QUARTER : sort by center : for(k=0,2,1,3;k<4;) 1+k,n-k,4+k,n-4-k,...
--OCTA : sort by center : for(k=0,4,2,6,1,5,3,7;k<8;) 1+k,n-k,8+k,n-8-k,...
--BISECT : sort by bisector (default) : 0,n-1,n/2,n/4,n-1-n/4,...
--seq : sequential order as of v2d
--debug : debug mode. not run sbatch
--force : vex2difx --force
-PCAL : vex2difx -d -f
--nohup : nohup ./run.EXP.CORRbySCAN >& log.run &
--opt <option between double quote>
--nmax <n>
```

‘run.EXP.CORRbySCAN’ 스크립트는 v2d 파일에 등록되어 있는 스캔 목록을 저장되어 있는 순서대로 만든다. 대개의 경우 순차적으로 데이터를 처리하므로, 전체 경향과 결과를 확인하는데 많은 시간을 기다려야 한다. 이러한 대기 시간을 단축하기 위하여 데이터 처리 순서를 ‘bisect’ 방식으로 분할하여, 전체 경향과 결과를 빠르게 예측할 수 있도록 상관처리 순서를 섞는 방안을 강구하였다. 이러한 상관처리를 위하여 스캔 순서를 섞어야 하는데, 디폴트 방식으로 사용하는 것이 ‘BISECT’ 모드이다. ‘BISECT’ 모드는 ‘1, n, n/2’ 와 같은 방법으로 스캔 순서를 섞는 방법이며, 이렇게 섞는 기능을 위하여 ‘bisector.py’ 파이썬 유틸리티를 개발하였다.

```
bisectory.py $max_number_of_scan
```

‘bisector.py’에 최대 스캔 개수를 입력하면 이를 섞을 인덱스를 출력한다. 이 출력 결과물을 스캔 순서 인덱스로 사용하면 가장 효율적으로 스캔을 섞어, 상관처리가 진행되면서 중간 결과를 확인할 때, 전체 평균과 경향을 빠른 시간에 파악할 수 있다.

스캔 개수가 ‘10’일 경우, 처리되는 스캔 순서는 ‘0 9 5 2 7 1 8 3 6 4’이며, 실제 출력된 결과는 다음과 같다.

```
[oper@coma FS]$ bisector.py 10  
0 9 5 2 7 1 8 3 6 4
```

표준전파원으로 등록된 스캔을 모두 처리하는 것이 가장 바람직하지만, 어떤 경우에는 적당한 개수에서 표준전파원 처리를 완료할 필요가 있다. 이러한 경우를 대비하여 ‘run.EXP.CORRbySCAN’에 ‘--nmax’ 옵션을 추가하였다.

```
./run.EXP.CORRbySCAN --nmax 5
```

와 같이 사용하면, 최대 스캔갯수를 ‘5’개로 제한하여 처리한다.

물론 스캔 순서를 섞는 방법도 위의 help에서 알 수 있듯이 따로 지정할 수 있다. 만약, 스캔 순서를 순차적으로 하기를 원한다면 ‘--seq’ 옵션을 사용하면 되며, 사용 방법은 다음과 같다.

```
./run.EXP.CORRbySCAN --seq
```

나. v2d 파일 자동 업데이트

프린지 상관처리가 완료될 때까지 기다리는 ‘wait_till_corr.py’ 스크립트는

–상관처리를 완료될 때까지 기다리고

–‘run.difx’을 스크립트를 수행시켜 ‘difx2fits’을 실행시켜 fits 결과를 얻는다.

‘run.PARSE.apd_delays.sh’ 스크립트를 실행하면, ‘log.apd.time’ 파일을 작성하고

‘apddelay_util.py’ 파이썬 유틸리티를 수행시켜 ‘log,{기선}.chan.xx’ 파일을 만든다.

‘gnuplot gnuplotFS.dem’을 실행시키면, 처리된 결과를 그림으로 결과를 출력시키고

‘run.clockrate.sh’를 실행시켜 clockrate와 clock offset 결과를 파일로 작성하며,

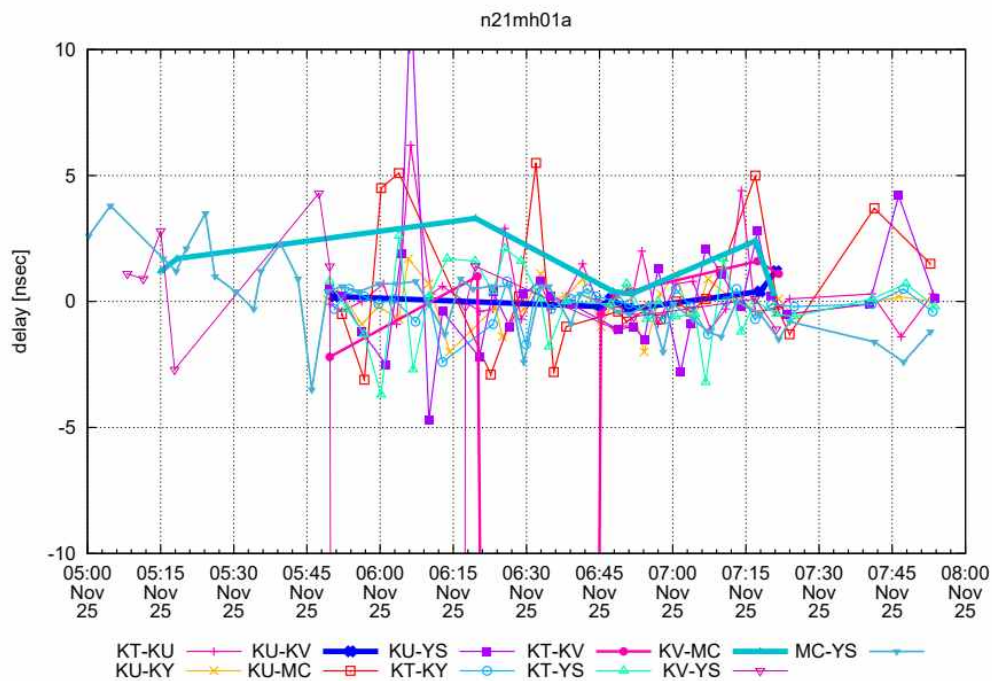
‘update.clockrate.sh’를 실행시켜 v2d 파일에 해당 관측사이트의 clockrate와 clock offset 값을 수정하여 준다.

표준전파원으로 얻어지는 ‘clockrate’가 만족되면, ‘checkall.sh’을 실행시킨다.

‘checkall.sh’ 스크립트는

- ‘all’이라는 디렉토리를 만들고
- 표준전파원을 이용한 처리결과인 v2d파일 등을 ‘all’ 디렉토리로 복사하며,
- 최종 수정된 v2d파일을 이용하여 상관처리가 진행되도록 vex 파일의 관측 \$MODE; 블록을 수정하고
- v2d 파일의 관측시작과 끝에 해당된 ‘mjdStart’와 ‘mjdStop’의 시간을 관측 vex파일에 있는 관측 시작과 끝 시간으로 수정하며,
- ‘run.vex’을 실행시켜 상관처리에 필요한 파일들을 생성시킨 다음
- ‘run.sbatch’를 실행시켜 상관처리가 진행되도록 자동 처리한다.

‘checkall.sh’는 상관처리를 요청한 다음, ‘wait_till_corr.py’ 파이썬 유틸리티를 수행시키고 종료된다. ‘wait_till_corr.py’ 스크립트는 상관처리가 완료될 때까지 기다리다가, 상관처리가 완료되면 ‘run.difx’를 실행시켜 최종 fits 결과를 생성한다. 아래 그림은 산출된 clockrate과 delay 결과를 날짜별로 누적해서 보여줌으로써, 안정적인 clock이 유지되고 있는지 확인할 수 있다.



<그림 3-1> n21mh01a에 대한 clockrate 및 delay 처리결과. 참여 사이트는 세종+KVN+MC(이탈리아)+YS(스페인).

4. VLBI 관측 데이터의 저장·관리·처리 자동화를 위한 관련 유틸리티 기능 업그레이드

본 연구의 1차년도에서는 세종 우주측지관측센터와 KVN 사이의 데이터 전송을 위한 globus-url-copy 프로토콜을 이용한 전송 방법을 제시하였다. VLBI 관측에서는 매년 수백 개의 관측을 수행하고 있다. 한 관측에는 4개의 관측소(세종 + 연세, 울산, 탐라)가 존재하게 되며 이들 데이터를 관측소별로 구별하게 되면 1000개 이상의 디렉토리가 존재하게 된다. 이러한 데이터 저장 구조는 관측코드와 관측소 이름으로 구분하여 저장하면 쉽게 구분할 수 있다. 또한 데이터 전송 유틸리티와 연계하여 약속된 일정한 장소에 저장하도록 함으로써 데이터 관리에 일관성과 편리성을 추구할 수 있다.

가. 데이터 전송 관리 기능 유틸리티 개발

데이터 전송은 마크6을 이용한 데이터와 마크5를 이용한 데이터를 구분하여 수행된다. 마크5 데이터는 대개 ‘1G 모드’를 이용한 관측을 의미하며 KVN에서는 이전 방식인 마크5B 레코더를 이용하여 기록하며, 기록하는 매체를 디스크팩이라 부르고 있다. 디스크팩은 고유의 번호를 갖도록 되어 있으며 ‘KVN-XXXX’와 같은 형식의 이름을 갖는다. 따라서 마크5 데이터는 전송된 다음 해당 디스크팩 이름으로 만들어진 디렉토리에 저장되도록 스크립트화되어 있다. 예를 들어 ‘KVN-4184’라는 디스크팩의 데이터를 전송하게 되면 ‘KVN-4184/DATA’라는 디렉토리를 만들고, 이 디렉토리에 데이터를 전송하게 된다.

한편 마크6 데이터는 대개 ‘8G 모드’와 ‘16G 모드’ 관측데이터를 의미하며 각 관측소(세종 + KVN)에서는 마크5와는 달리 고정된 불박이 방식의 디스크팩에 저장한다. 이러한 특성을 반영하여 마크6 데이터가 전송되면 ‘mk6-관측소’의 형식을 갖도록 하고 있다. 각 관측소에 설치된 마크6 레코더 시스템의 ‘.bashrc’ 파일에는 ‘export SITE_ID=mk6-XX’라는 설정이 되어 있으며, 이러한 ‘SITE_ID’를 이용하여 마크6 레코더 시스템을 구분한다. 세종의 경우 ‘mk6-KV’의 이름으로 설정되어 있으며, KVN에서는 ‘mk6-YS’, ‘mk6-US’, ‘mk6-TN’ 등으로 설정되어 있다. 해당 마크6에서 데이터 전송을 시작하면 대전 스토리지 서버에는 ‘mk6-KV/DATA’의 디렉토리가 만들어지고, 이 디렉토리에 세종 마크6의 데이터가 전송되어 저장된다.

마크6 데이터 전송을 시작하게 되면, 각 사이트의 마크6에서는 데이터 목록 파일인 ‘mk6-XX.lis’ 파일을 ‘mk6-XX’ 디렉토리에 전송한다. 이 디렉토리에서 ‘fuse2dir.py’ 파일인 유틸리티를 실행시키면 현재 디렉토리에서 ‘.lis’ 파일을 검색한 다음 디렉토리 이름

‘mk6-XX’에 해당되는 ‘mk6-XX.lis’ 파일을 알아낸다. ‘fuse2dir.py’은 관측코드별 파일 이름과 크기별로 정리한 다음 그 결과를 ‘copy.info’ 파일에 저장한다. 이 ‘copy.info’은 전송할 데이터의 내용을 브리핑하는 역할을 하므로, 데이터 전송의 전체 내용과 크기를 파악하는데 편리하도록 개발되었다.

파일 ‘copy.info’의 내용을 살펴보면 다음과 같다.

# mk6-YS2				
eb092e	KY	661 GB	709,295,069,363	0:11:29
a2225d	KY	18,009 GB	19,336,624,355,474	5:13:29
a2225e	KY	18,295 GB	19,643,502,754,656	5:18:28
gp060	KY	7,283 GB	7,819,215,753,967	2:06:46
eb092f	KY	4,797 GB	5,149,817,528,292	1:23:29
t22jp02a	KY	8,439 GB	9,060,845,919,403	2:26:54
t22db01a	KY	2,298 GB	2,466,938,823,211	0:39:59
n22kh01a	KY	76,594 GB	82,241,114,359,134	22:13:21
s22tj01a	KY	12,038 GB	12,925,310,422,700	3:29:33
f22315a	KY	1,149 GB	1,233,402,400,175	0:19:59
n22hk01c	KY	22,064 GB	23,690,581,081,020	6:24:05
n22kn02b	KY	38,603 GB	41,449,402,869,780	11:12:00

마크6 데이터 전송에 사용되고 ‘copy.info’ 파일 작성에 사용되는 ‘mk6-YS.lis’ 파일의 내용은 다음과 같다.

t22jp02a_KVNYS_313204233.vdif	616,722,480,916	2022-11-10 05:42
t22jp02a_KVNYS_313204822.vdif	426,637,605,779	2022-11-10 05:48
t22jp02a_KVNYS_313205234.vdif	308,241,334,479	2022-11-10 05:52
t22jp02a_KVNYS_313210826.vdif	308,238,069,552	2022-11-10 06:08
...		
t22db01a_KVNYS_314171700.vdif	616,710,523,209	2022-11-11 02:17
n22kh01a_KVNYS_314182500.vdif	493,428,585,088	2022-11-11 03:25
n22kh01a_KVNYS_314182930.vdif	1,726,674,082,203	2022-11-11 03:29
...		
n22kh01a_KVNYS_315060440.vdif	2,444,653,937,522	2022-11-11 15:04
s22tj01a_KVNYS_315080515.vdif	592,137,630,304	2022-11-11 17:05
...		
s22tj01a_KVNYS_315105300.vdif	616,492,439,088	2022-11-11 19:53
f22315a_KVNYS_315162300.vdif	616,628,297,200	2022-11-12 01:23
f22315a_KVNYS_315163900.vdif	616,774,102,975	2022-11-12 01:39
n22hk01c_KVNYS_317152729.vdif	1,210,893,231,712	2022-11-14 00:27
...		
n22hk01c_KVNYS_318005218.vdif	431,989,005,853	2022-11-14 09:52
n22kn02b_KVNYS_318094500.vdif	616,800,000,000	2022-11-14 18:45
...		
n22kn02b_KVNYS_318213845.vdif	385,499,323,800	2022-11-15 06:38

나. 데이터 저장 기능 유틸리티 개발

데이터 전송이 관측코드별로 완료가 되면, 전송된 데이터를 상관계리에 편리하도록 분류하고 저장하여야 한다.

먼저, 전송된 데이터를 관측코드별로 분류하려면 ‘parkingDATA.sh’ 스크립트를 실행시킨다. ‘parkingDATA.sh’을 실행시키면 ‘DATA/’ 디렉토리에 저장되어 있는 데이터를 관측코드별로 분류하고 분류된 데이터를 관측코드별 디렉토리를 생성하고 이 디렉토리에 해당 데이터를 이동시킨다. 데이터를 저장할 디렉토리는

{관측코드}-{사이트이름}-{데이타포맷}

와 같은 형식으로 만들어진다.

위의 ‘mk6-YS.lis’ 파일과 데이터 전송 요약을 알려주는 ‘copy.info’을 참조해 보면 ‘t22jp02a’ 관측코드에 해당되는 데이터는 ‘t22jp02a-KY-VDIF’ 디렉토리에 이동되어 저장된다.

관측코드별 디렉토리에 저장된 데이터는 ‘chkpspeed’ 유틸리티를 이용하여 데이터 정보를 저장하는데, 이 경우에는 ‘vdif.yyyy-mm-dd.hhmmss.mk6-YS.t22jp02a-KY-VDIF’라는 파일에 다음과 같이 기록된다,

2022-11-17	5:52:53	308,398,560,800	t22jp02a_KVNYS_313195103.vdif	2297.75	Gbsec	
2022-11-17	5:58:59	616,722,233,856	t22jp02a_KVNYS_313204233.vdif	4594.94	Gbsec	12.554 Gbps
2022-11-17	6:08:06	426,637,434,880	t22jp02a_KVNYS_313204822.vdif	3178.70	Gbsec	5.811 Gbps
2022-11-17	6:10:01	616,713,861,824	t22jp02a_KVNYS_313195400.vdif	4594.88	Gbsec	39.955 Gbps
2022-11-17	6:14:53	308,241,235,952	t22jp02a_KVNYS_313205234.vdif	2296.58	Gbsec	7.865 Gbps
2022-11-17	6:21:24	308,237,946,080	t22jp02a_KVNYS_313210826.vdif	2296.55	Gbsec	5.874 Gbps
2022-11-17	6:28:38	308,390,133,744	t22jp02a_KVNYS_313211525.vdif	2297.69	Gbsec	5.294 Gbps
2022-11-17	6:29:33	616,645,602,624	t22jp02a_KVNYS_313200120.vdif	4594.37	Gbsec	83.534 Gbps
2022-11-17	6:32:44	308,320,350,560	t22jp02a_KVNYS_313203214.vdif	2297.17	Gbsec	12.027 Gbps
2022-11-17	6:35:44	308,328,179,808	t22jp02a_KVNYS_313212035.vdif	2297.22	Gbsec	12.762 Gbps
2022-11-17	6:40:36	308,397,203,840	t22jp02a_KVNYS_313200648.vdif	2297.74	Gbsec	7.869 Gbps
2022-11-17	6:46:00	616,796,570,592	t22jp02a_KVNYS_313202155.vdif	4595.49	Gbsec	14.184 Gbps
2022-11-17	6:52:01	616,790,912,480	t22jp02a_KVNYS_313212337.vdif	4595.45	Gbsec	12.730 Gbps
2022-11-17	6:55:55	308,393,593,504	t22jp02a_KVNYS_313201138.vdif	2297.71	Gbsec	9.819 Gbps
2022-11-17	6:57:48	616,794,613,280	t22jp02a_KVNYS_313203513.vdif	4595.48	Gbsec	40.668 Gbps
2022-11-17	6:58:15	308,377,688,288	t22jp02a_KVNYS_313202724.vdif	2297.59	Gbsec	85.096 Gbps
2022-11-17	7:06:11	616,792,178,976	t22jp02a_KVNYS_313213057.vdif	4595.46	Gbsec	9.654 Gbps
2022-11-17	7:13:11	616,794,506,368	t22jp02a_KVNYS_313201435.vdif	4595.48	Gbsec	10.942 Gbps
2022-11-17	7:13:30	308,284,025,152	t22jp02a_KVNYS_313213952.vdif	2296.89	Gbsec	120.889 Gbps
2022-11-17	7:19:55	308,394,901,120	t22jp02a_KVNYS_313214651.vdif	2297.72	Gbsec	5.968 Gbps
2022-11-17	7:26:01	308,390,632,864	t22jp02a_KVNYS_313215431.vdif	2297.69	Gbsec	6.278 Gbps
8,438 GiB (9,060,842,366,592 B) 1:36:25 5,251 GiB/h 0:11:41/TiB						

<V>= 11,949 Mibps(1,493 MiB/sec = 87.5 GiB/min) <V/Vo>= 1166% <t/to>= 0.086

이 로그 정보는 나중에 해당 ‘mk6-XX/HISTORY’ 디렉토리에 다시 저장되어 데이터 전송과 년차별 데이터 전송 현황과 용량 조사에 사용된다.

디렉토리별로 저장된 데이터를 DIFX 상관처리를 위하여 적절한 디렉토리로 이동시켜야 하는데 이를 위하여서는 ‘move.DATA’ 스크립트를 이용한다.

```
move.DATA t22jp02a-KY-VDIF
```

위와 같이 입력하면, ‘t22jp02a-KY-VDIF’ 디렉토리를 상관처리용 데이터를 보관하는 스토리지의 ‘DATA’ 디렉토리에 관측코드에 해당되는 ‘t22jp02a’ 디렉토리를 만들고, 이 디렉토리 안에 ‘t22jp02a-KY-VDIF’ 데이터를 이동시킨다. 관측코드별로 디렉토리가 만들어지고, 이 디렉토리 안에 해당 사이트별 관측데이터가 디렉토리별로 분류되어 저장된다.

DIFX 상관처리를 위해, 어떤 데이터가 있는지 filelist를 만들어야 한다. 스크립트 ‘run.filelist’를 수행하면 현재 디렉토리를 포함한 경로(PATH) 정보에서 관측코드를 추출하여, 추출된 관측코드가 스토리지 ‘DATA/{관측코드}’ 디렉토리가 있는지 검사한다. 관측코드 디렉토리가 존재하고, 디렉토리 내부에 관측코드별 사이트 디렉토리가 존재하면 사이트와 데이터 포맷별로 filelist를 생성한다. 위의 예에서 사용한 ‘t22jp02a-KY-VDIF’ 디렉토리는 ‘filelist.KY-VDIF’라는 파일에 저장된다. 생성된 filelist 파일의 내용은 다음과 같다.

/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313195103.vdif	2022-11-09T19:51:03s	2022-11-09T19:53:32s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313195400.vdif	2022-11-09T19:54:00s	2022-11-09T19:58:59s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313200120.vdif	2022-11-09T20:01:20s	2022-11-09T20:06:19s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313200648.vdif	2022-11-09T20:06:48s	2022-11-09T20:09:17s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313201138.vdif	2022-11-09T20:11:38s	2022-11-09T20:14:07s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313201435.vdif	2022-11-09T20:14:35s	2022-11-09T20:19:34s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313202155.vdif	2022-11-09T20:21:55s	2022-11-09T20:26:54s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313202724.vdif	2022-11-09T20:27:24s	2022-11-09T20:29:53s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313203214.vdif	2022-11-09T20:32:14s	2022-11-09T20:34:43s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313203513.vdif	2022-11-09T20:35:13s	2022-11-09T20:40:12s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313204233.vdif	2022-11-09T20:42:33s	2022-11-09T20:47:32s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313204822.vdif	2022-11-09T20:48:24s	2022-11-09T20:51:51s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313210826.vdif	2022-11-09T21:08:26s	2022-11-09T21:10:55s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313212035.vdif	2022-11-09T21:20:35s	2022-11-09T21:23:04s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313212337.vdif	2022-11-09T21:23:37s	2022-11-09T21:28:36s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313213057.vdif	2022-11-09T21:30:57s	2022-11-09T21:35:56s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313213952.vdif	2022-11-09T21:39:52s	2022-11-09T21:42:21s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313214651.vdif	2022-11-09T21:46:51s	2022-11-09T21:49:20s
/MD3460/mk5/DATA/t22jp02a/t22jp02a-KY-VDIF/t22jp02a_KVNYS_313215431.vdif	2022-11-09T21:54:31s	2022-11-09T21:57:00s

제4장

e-VLBI 관측 안정화를 위한 성능 평가

-
1. DiFX 상관이 성능 개선을 위한 서버 추가 연동,
DiFX 상관이 SW 업데이트 등 관측환경 조성
 2. e-VLBI 성능 업그레이드, DiFX 상관을 이용한
e-VLBI 상관처리 테스트
 3. e-VLBI 운용을 위한 사용자 매뉴얼 제작 및 e-VLBI 활용방안 제시

1. DiFX 상관계 성능 개선을 위한 서버 추가 연동, DiFX 상관계 SW 업데이트 등 관측 환경 조성

가. DiFX 서버 환경 분석

1차년도 사업으로부터 세종 망원경에서 사용하고 있는 DiFX 클러스터를 이용하여 세종-KVN 4기의 전파망원경이 참여하는 1024 Mbps 실시간 e-VLBI 관측 및 상관처리를 성공적으로 수행하였다. 세종 DiFX 클러스터로 이용하고 있는 2기의 서버는 belbi1과 belbi2이며, 다음과 같은 사양을 갖고 있다.

(1) belbi1 (old cluster server): Dell PowerEdge R510

서버 (총 7개 노드)

- 2CPU / Quad-core Xeon 54500, 4 and 8 MB
- 32GB, 현재 6개 노드만 이용 (96 cores available)
- OS: Linux version

2.6.32-642.13.1.el6.centos.plus.x86_64

(mockbuild@c1bm.rdu2.centos.org)

(gcc version 4.4.7 20120313 (Red Hat 4.4.7-17)

(GCC)) #1 SMP Thu Jan 12 11:45:05 UTC 2017

(2) belbi2 (new cluster server): Dell PowerEdge R440

서버 (총 3개 노드)

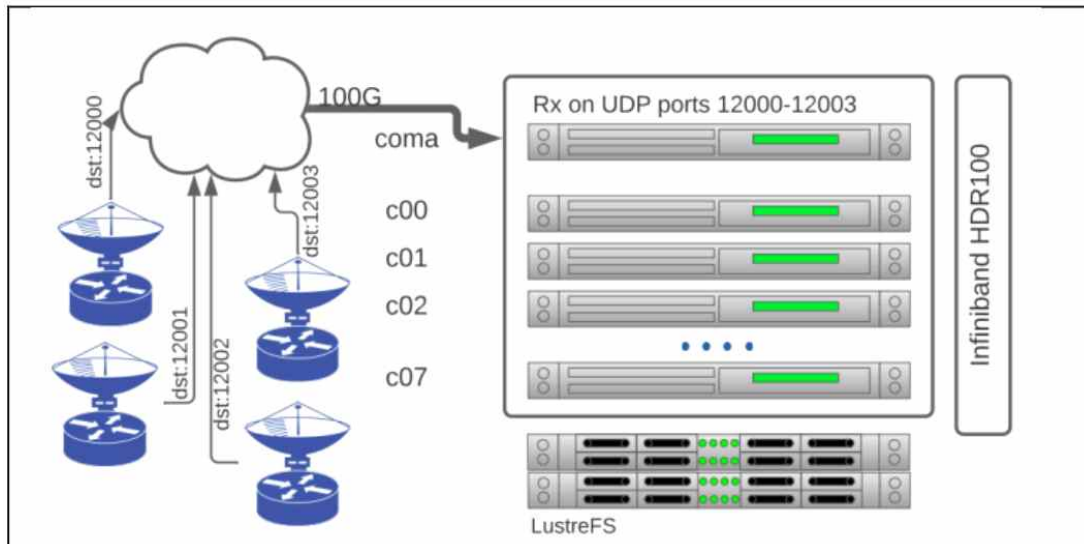
- 2CPU / Intel Xeon Silver 4214R, 2.4G, 12C/24T,
9.6GT/s, 16.5M 캐시
- 32GB RDIMM, 3200MT/s, 듀얼 랙크

- OS: Linux version 3.10.0-1160.6.1.el7.x86_64

(mockbuild@kbuilder.bsys.centos.org)

(gcc version 4.8.5 20150623 (Red Hat 4.8.5-44)

(GCC)) #1 SMP Tue Nov 17 13:59:11 UTC 2020



<그림 4-1> e-VLBI를 위한 클러스터와 스토리지 구성 개념도(예시)



<그림 4-2> 세종 DiFX belbi2 서버.

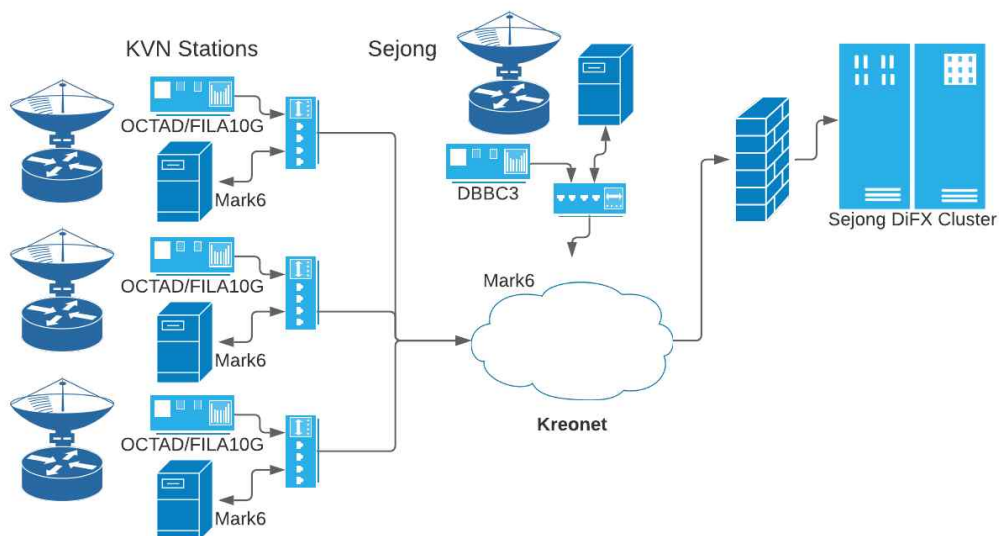
기존 belbi2 3기와 동일한 노드 3기가 추가되어 총 6개로 확장됨.

belbi1의 경우 기존에 사용하고 있던 서버이며, belbi2가 1차년도 사업당시 도입된 서버 클러스터이다. 1차년도 사업에서 파악한 belbi1과 belbi2의 성능분석 결과 및 e-VLBI를 위한 belbi DiFX 서버 개선 방안에서 기존의 belbi1 서버와 belbi2 서버를 병합하여 운영하는 방안을 비롯하여 belbi2 클러스터를 성능 확장하는 방법 등을 제시하였다. 2차년도에는 belbi2와 동일한 사양을 갖는 클러스터 노드 3기가 세종에서 추가하였고, 새롭게 추가된 3개의 노드를 서버를 병합하여 단일 서버로 구성하고, 통합된 belbi2 클러스터에 DiFX 소프트웨어를 설치하였다.

나. DiFX belbi2 환경 설정 및 테스트

세종 belbi2 6개 노드로 구성된 DiFX 서버에 e-VLBI 상관처리에 필요한 환경을 설정하고, 성공적으로 테스트를 완료하였다. 단, 해당 내용에는 belbi2 서버의 세부 내용(아이피, 포트, 네트워크 설정 등)이 모두 담겨 있는 중요 보안 사항으로 본 보고서에는 넣지 않고 별도의 파일로 제출한다.

2. e-VLBI 성능 업그레이드 DiFX 상관기를 이용한 e-VLBI 상관처리 테스트



<그림 4-3> 세종-KVN 실시간 e-VLBI 구성도

가. 세종 + KVN 2Gbps e-VLBI 관측 설정

세종 belbi2 DiFX 서버를 이용하여 1차년도 1Gbps e-VLBI 관측 성공에 이어, 2차년도에는 다음과 같이 2Gbps e-VLBI 상관처리를 테스트하였다. 이 관측은 세종과 한국천문연구원 운영하고 있는 한국우주전파관측망(KVN) 3기의 전파망원경을 이용하여 수행하였으며, 2Gbps e-VLBI 관측 vex 파일 설정은 다음과 같다.

```
VEX_rev = 1.5;
* SCHED vers: Development version 11.5. Started Apr. 2
*-----
$CLOCK;
def Kt;
clock_early = 2022y096d04h35m00s : -1.7728 usec : 2022y311d02h00m00s : +2.2534e-12; enddef;
def Ku;
clock_early = 2022y096d04h35m00s : 0.0 usec : 2022y311d02h00m00s : 0.0e-12;enddef;
def Ky;
clock_early = 2022y096d04h35m00s : -1.1688 usec : 2022y311d02h00m00s : +2.3938e-12; enddef;
def Kv;
clock_early = 2022y096d04h35m00s : -25.6181 usec : 2022y311d02h00m00s : +2.5909e-12; enddef;
*-----
$MODE;
*
def kq;
ref $PROCEDURES = NoProc:Ky:Ku:Kt:Kv;
ref $FREQ = FreqOCTAD:Kt:Ku:Ky;
ref $FREQ = FreqDBBC3:Kv;
ref $IF = StdIFs:Ky:Kt:Ku:Ky;
ref $BBC = StdBBCs:Kv:Kt:Ku:Ky;
ref $TRACKS = VDIF.8224:Kt:Ku:Ky:Kv;
ref $ROLL = NoRoll:Kt:Ku:Ky:Kv;
ref $PHASE_CAL_DETECT = NoDetect:Kt:Ku:Ky:Kv;
enddef;
*-----
$PROCEDURES;
*
def NoProc;
* pass
enddef;
*-----
$STATION;
*
def Kt;
ref $SITE = KVNTN;
ref $ANTENNA = KVNTN;
ref $DAS = 2NONE+MARK5B<;
* ref $DAS = MARK6;
enddef;
*
def Ku;
ref $SITE = KVNUS;
ref $ANTENNA = KVNUS;
ref $DAS = 2NONE+MARK5B<;
* ref $DAS = MARK6;
enddef;
*
def Ky;
ref $SITE = KVNYS;
ref $ANTENNA = KVNYS;
ref $DAS = 2NONE+MARK5B<;
* ref $DAS = MARK6;
enddef;
*
def Kv;
ref $SITE = SEJONG;
ref $ANTENNA = SEJONG;
ref $DAS = MARK6;
enddef;
*-----
$SITE;
*
def KVNTN;
site_type = fixed;
site_name = KVNTN;
site_ID = Kt;
* n22hy01a used KaVA K-band geodesy (28 Jan 2014, MJD 56685) : updated at 06 Sep 2014
* X= -3171731.5532 Y=4292678.5258 Z=3481038.7679
site_position =-3171731.55320 m: 4292678.52580 m: 3481038.76790 m;
```

```

site_velocity = 0.000000 m/yr: 0.000000 m/yr: 0.000000 m/yr;
* First line below is VEX standard format. Use only when readers are ready.
* site_position_epoch = 2014y028d;
site_position_epoch = 56685;
horizon_map_az = 0.0 deg: 35.0: 39.0: 45.0: 50.0: 55.0: 60.0: 70.0: 80.0:
90.0:295.0:310.0:315.0:320.0:330.0:335.0:345.0;
horizon_map_el = 9.0 deg: 8.0: 9.0: 8.0: 7.0: 6.0: 5.0: 4.0: 3.0:
2.0: 3.0: 4.0: 5.0: 6.0: 7.0: 8.0: 9.0;
enddef;
*
def KVNUS;
site_type = fixed;
site_name = KVNUS;
site_ID = Ku;
* n22hy01a used KaVA K-band geodesy (28 Jan 2014, MJD 56685) : updated at 06 Sep 2014
* X=-3287268.5430 Y=4023450.1448 Z=3687379.9675
site_position =-3287268.54300 m: 4023450.14480 m: 3687379.96750 m;
site_velocity = 0.000000 m/yr: 0.000000 m/yr: 0.000000 m/yr;
* First line below is VEX standard format. Use only when readers are ready.
* site_position_epoch = 2014y028d;
site_position_epoch = 56685;
horizon_map_az = 0.0 deg: 5.0: 10.0: 20.0: 30.0: 35.0: 40.0:175.0:180.0:
185.0:190.0:195.0:200.0:205.0:210.0:215.0:230.0:
235.0:248.0:255.0:265.0:270.0:280.0:285.0:290.0:
305.0:310.0:337.0:345.0:350.0;
horizon_map_el = 7.0 deg: 3.0: 7.0: 6.0: 2.0: 3.0: 2.0: 3.0: 4.0:
4.0: 5.0: 4.0: 5.0: 6.0: 7.0: 8.0: 9.0:
8.0: 9.0: 7.0: 6.0: 4.0: 5.0: 3.0: 2.0:
3.0: 2.0: 3.0: 2.0: 7.0;
enddef;
*
def KVNYS;
site_type = fixed;
site_name = KVNYS;
site_ID = Ky;
* n22hy01a used KaVA K-band geodesy (28 Jan 2014, MJD 56685) : updated at 06 Sep 2014
* X=-3042280.9035 Y=4045902.6564 Z=3867374.3087
site_position =-3042280.90350 m: 4045902.65640 m: 3867374.30870 m;
site_velocity = 0.000000 m/yr: 0.000000 m/yr: 0.000000 m/yr;
* First line below is VEX standard format. Use only when readers are ready.
* site_position_epoch = 2014y028d;
site_position_epoch = 56685;
horizon_map_az = 0.0 deg: 5.0: 10.0: 15.0: 25.0: 30.0: 45.0: 65.0: 75.0:
85.0:110.0:112.0:175.0:180.0:200.0:245.0:250.0;
horizon_map_el = 3.0 deg: 5.0: 7.0: 9.0: 8.0: 6.0: 7.0: 5.0: 4.0:
3.0: 6.0: 3.0: 6.0: 7.0: 3.0: 4.0: 3.0;
enddef;
*
def SEJONG;
site_type = fixed;
site_name = SEJONG;
site_ID = Kv;
* elev= 194.62 long=-127:18:12. lat= 36:31:21.8
site_position =-3110079.96000 m: 4082066.73400 m: 3775076.83200 m;
site_velocity = 0.000000 m/yr: 0.000000 m/yr: 0.000000 m/yr;
* First line below is VEX standard format. Use only when readers are ready.
* site_position_epoch = 2014y028d;
site_position_epoch = 56685;
horizon_map_az = 0.0 deg: 35.0: 39.0: 45.0: 50.0: 55.0: 60.0: 70.0: 80.0:
90.0:295.0:310.0:315.0:320.0:330.0:335.0:345.0;
horizon_map_el = 9.0 deg: 8.0: 9.0: 8.0: 7.0: 6.0: 5.0: 4.0: 3.0:
2.0: 3.0: 4.0: 5.0: 6.0: 7.0: 8.0: 9.0;
enddef;
*-----
$ANTENNA;
*
def KVNTN;
axis_type = az : el;
antenna_motion = el : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
antenna_motion = az : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
axis_offset = 0.00000 m;
enddef;
*
def KVNUS;
axis_type = az : el;
antenna_motion = el : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
antenna_motion = az : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
axis_offset = 0.00000 m;
enddef;
*
def KVNYS;
axis_type = az : el;
antenna_motion = el : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
antenna_motion = az : 177.0 deg/min : 2 sec: * 3.000 deg/sec/sec
axis_offset = 0.00000 m;

```



```

endif;
*
def SEJONG;
axis_type = az : el;
antenna_motion = el : 177.0 deg/min : 2 sec; * 5.000 deg/sec/sec
antenna_motion = az : 177.0 deg/min : 2 sec; * 5.000 deg/sec/sec
* pointing_sector = &ccw: az : -90 : 0 : el : 5 : 85 ;
* pointing_sector = &n : az : 0 : 360 : el : 5 : 85 ;
* pointing_sector = &cw : az : 360 : 450 : el : 5 : 85 ;
axis_offset = 0.00000 m;
endif;
*-----
$FREQ;
*
def FreqDBBC3; * Kv
sample_rate = 64.000 Ms/sec; * (2bits/sample)
chan_def = : 22044.00 MHz : U : 32.00 MHz : &CH01 : &BBC_L : &NoCal; *Lcp
chan_def = : 22044.00 MHz : L : 32.00 MHz : &CH02 : &BBC_L : &NoCal; *Lcp
chan_def = : 22108.00 MHz : U : 32.00 MHz : &CH03 : &BBC_L : &NoCal; *Lcp
chan_def = : 22108.00 MHz : L : 32.00 MHz : &CH04 : &BBC_L : &NoCal; *Lcp
chan_def = : 22172.00 MHz : U : 32.00 MHz : &CH05 : &BBC_L : &NoCal; *Lcp
chan_def = : 22172.00 MHz : L : 32.00 MHz : &CH06 : &BBC_L : &NoCal; *Lcp
chan_def = : 22236.00 MHz : U : 32.00 MHz : &CH07 : &BBC_L : &NoCal; *Lcp
chan_def = : 22236.00 MHz : L : 32.00 MHz : &CH08 : &BBC_L : &NoCal; *Lcp
chan_def = : 22300.00 MHz : U : 32.00 MHz : &CH09 : &BBC_L : &NoCal; *Lcp
chan_def = : 22300.00 MHz : L : 32.00 MHz : &CH10 : &BBC_L : &NoCal; *Lcp
chan_def = : 22364.00 MHz : U : 32.00 MHz : &CH11 : &BBC_L : &NoCal; *Lcp
chan_def = : 22364.00 MHz : L : 32.00 MHz : &CH12 : &BBC_L : &NoCal; *Lcp
chan_def = : 22428.00 MHz : U : 32.00 MHz : &CH13 : &BBC_L : &NoCal; *Lcp
chan_def = : 22428.00 MHz : L : 32.00 MHz : &CH14 : &BBC_L : &NoCal; *Lcp
chan_def = : 22492.00 MHz : U : 32.00 MHz : &CH15 : &BBC_L : &NoCal; *Lcp
chan_def = : 22492.00 MHz : L : 32.00 MHz : &CH16 : &BBC_L : &NoCal; *Lcp
endif;
*
def FreqOCTAD; * KVN
sample_rate = 64.000 Ms/sec; * (2bits/sample)
chan_def = : 22044.00 MHz : U : 32.00 MHz : &CH01 : &BBC_L : &NoCal; *Lcp
chan_def = : 22044.00 MHz : L : 32.00 MHz : &CH02 : &BBC_L : &NoCal; *Lcp
chan_def = : 22108.00 MHz : U : 32.00 MHz : &CH03 : &BBC_L : &NoCal; *Lcp
chan_def = : 22108.00 MHz : L : 32.00 MHz : &CH04 : &BBC_L : &NoCal; *Lcp
chan_def = : 22172.00 MHz : U : 32.00 MHz : &CH05 : &BBC_L : &NoCal; *Lcp
chan_def = : 22172.00 MHz : L : 32.00 MHz : &CH06 : &BBC_L : &NoCal; *Lcp
chan_def = : 22236.00 MHz : U : 32.00 MHz : &CH07 : &BBC_L : &NoCal; *Lcp
chan_def = : 22236.00 MHz : L : 32.00 MHz : &CH08 : &BBC_L : &NoCal; *Lcp
chan_def = : 22300.00 MHz : U : 32.00 MHz : &CH09 : &BBC_L : &NoCal; *Lcp
chan_def = : 22300.00 MHz : L : 32.00 MHz : &CH10 : &BBC_L : &NoCal; *Lcp
chan_def = : 22364.00 MHz : U : 32.00 MHz : &CH11 : &BBC_L : &NoCal; *Lcp
chan_def = : 22364.00 MHz : L : 32.00 MHz : &CH12 : &BBC_L : &NoCal; *Lcp
chan_def = : 22428.00 MHz : U : 32.00 MHz : &CH13 : &BBC_L : &NoCal; *Lcp
chan_def = : 22428.00 MHz : L : 32.00 MHz : &CH14 : &BBC_L : &NoCal; *Lcp
chan_def = : 22492.00 MHz : U : 32.00 MHz : &CH15 : &BBC_L : &NoCal; *Lcp
chan_def = : 22492.00 MHz : L : 32.00 MHz : &CH16 : &BBC_L : &NoCal; *Lcp
endif;
*-----
$BBC;
*
def StdBBCs;
BBC_assign = &BBC_L : 1 : &IF_L;
BBC_assign = &BBC_R : 2 : &IF_R;
endif;
*-----
$IF;
*
def StdIFs;
if_def = &IF_L : A : L : 31200.0 MHz : U : 200 MHz : 0 Hz;
if_def = &IF_R : B : R : 31200.0 MHz : U : 200 MHz : 0 Hz;
endif;
*-----
$TRACKS;
def VDIF.8224;
track_frame_format = VDIF/8224/2;
endif;
def VDIF.8032;
track_frame_format = VDIF/8032/2;
endif;
def Mark5B.16CH;
* Format required for v2d 'fake=true' stations
track_frame_format = MARK5B;
fanout_def = : &CH01 : sign : 1: 2;
fanout_def = : &CH01 : mag : 1: 3;
fanout_def = : &CH02 : sign : 1: 4;
fanout_def = : &CH02 : mag : 1: 5;
fanout_def = : &CH03 : sign : 1: 6;
fanout_def = : &CH03 : mag : 1: 7;

```

```

fanout_def = : &CH04 : sign : 1: 8;
fanout_def = : &CH04 : mag : 1: 9;
fanout_def = : &CH05 : sign : 1: 10;
fanout_def = : &CH05 : mag : 1: 11;
fanout_def = : &CH06 : sign : 1: 12;
fanout_def = : &CH06 : mag : 1: 13;
fanout_def = : &CH07 : sign : 1: 14;
fanout_def = : &CH07 : mag : 1: 15;
fanout_def = : &CH08 : sign : 1: 16;
fanout_def = : &CH08 : mag : 1: 17;
fanout_def = : &CH09 : sign : 1: 18;
fanout_def = : &CH09 : mag : 1: 19;
fanout_def = : &CH10 : sign : 1: 20;
fanout_def = : &CH10 : mag : 1: 21;
fanout_def = : &CH11 : sign : 1: 22;
fanout_def = : &CH11 : mag : 1: 23;
fanout_def = : &CH12 : sign : 1: 24;
fanout_def = : &CH12 : mag : 1: 25;
fanout_def = : &CH13 : sign : 1: 26;
fanout_def = : &CH13 : mag : 1: 27;
fanout_def = : &CH14 : sign : 1: 28;
fanout_def = : &CH14 : mag : 1: 29;
fanout_def = : &CH15 : sign : 1: 30;
fanout_def = : &CH15 : mag : 1: 31;
fanout_def = : &CH16 : sign : 1: 32;
fanout_def = : &CH16 : mag : 1: 33;
enddef;
*-----
$SOURCE;
*
def 3C279;
source_name = 3C279;
* GSFC 2011B astro solution 7402 Observations
ra = 12h56m11.1665670s; dec = -05d47'21.524810"; ref_coord_frame = J2000;
enddef;
*
def NRAO530;
source_name = NRAO530;
* rfc_2012b Petrov, 2012, unpublished 17851 observations
ra = 12h56m11.1665670s; dec = -05d47'21.524810"; ref_coord_frame = J2000;
enddef;
*
def OJ287;
source_name = OJ287;
* rfc_2012b Petrov, 2012, unpublished 191510 observations
ra = 08h54m48.8749300s; dec = 20d06'30.640780"; ref_coord_frame = J2000;
enddef;
*
def 3C454.3;
source_name = 3C454.3;
* this source had calibrator code: V
* alternate source name: J2253+1608
* alternate source name: 2251+158
* alternate source name: J2253+16
* GSFC 2011B astro solution 39747 Observations
ra = 22h53m57.7479370s; dec = 16d08'53.560930"; ref_coord_frame = J2000;
* ra = 22h51m29.5197375s; dec = 15d52'54.348102"; ref_coord_frame = B1950;
* ra = 22h55m02.0560632s; dec = 16d15'44.524018"; ref_coord_frame = Date;
enddef;
*-----
$SCHED;
* schedule section for experiment s22tj01b
* KVN+Sejong Realtime e-VLBI 2Gbps Test Experiment
scan No0001;
start=2022y112d00h00m00s; mode=kq; source=3C454.3;
* :data_good:data_stop:goto_foot: pass: wrap :driv:tape at
station=Ky: 0 sec: 1800 sec: 0.000 GB: : : 1;
station=Ku: 0 sec: 1800 sec: 0.000 GB: : : 1;
station=Kt: 0 sec: 1800 sec: 0.000 GB: : : 1;
station=Kv: 0 sec: 1800 sec: 0.000 GB: : : 1;
endscan;
... 이후 스캔 정보는 길이상 생략

```

나. 세종 + KVN 2Gbps e-VLBI 상관처리를 위한 DiFX 설정

다음은 세종 belbi2 DiFX를 이용한 2Gbps e-VLBI 상관처리를 위한 .v2d 파일 설정이다. 4기의 전파망원경에서 산출되는 2Gbps 데이터 스트림을 실시간으로 받아 상관처리 하는 것

으로 총 8Gbps 스트림의 데이터를 실시간 상관처리 한다. 따라서 상관처리를 위한 4개의 thread를 정의하고, belbi2 cluster의 6개 노드, 총 53개 코어를 사용하도록 설정하였다. 이후 각 망원경의 데이터 스트림 주소와 포트 번호, UDP 패킷의 MTU를 설정하였다. 또한 수송원자시계와 GNSS의 시각 차이(clock offset) 값도 설정하였다. 망원경 수신기에서 검출된 신호가 디지털화 되는 백엔드 장비의 경우, 세종은 DBBC3를 이용하였으며, KVN은 OCTAD를 사용하였고, 각 장비에서 VIDF 포맷의 2Gbps 데이터가 세종 belbi2 DiFX에 정의된 포트로 실시간으로 전송되도록 설정하였다. 참고로 포트 번호는 보안상 xxx로 표기하였다.

```
dataBufferFactor = 8
visBufferLength = 32
startSeries = 1000
# exhaustiveAutocorr = True # supported in difx 2.5.3 and newer
minSubarray = 2
singleScan = True
singleSetup = True

machines = belbi2, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6
nThread = 4
nCore = 53

antennas = Kv,Ku,Kt,Ky

ANTENNA Kv
{
datastreams = dsKv
phaseCalInt = 0
clockOffset = +27.0 # +25.6181
clockRate = -0.0000025909 # f22jp01a
clockEpoch = 2022y090d14h10m00s
}

ANTENNA Ku
{
datastreams = dsKu
phaseCalInt = 0
# April 2022 clocks
clockOffset = +42.0
clockRate = 0.0
clockEpoch = 2022y096d04h35m00s
}

ANTENNA Kt
{
datastreams = dsKt
phaseCalInt = 0
# April 2022 clocks
clockOffset = +1.7728 # n22hy01a 2022-04-06 offset plus 2022-04-21 delta
clockRate = -0.0000022534000000000000
clockEpoch = 2022y096d04h35m00s
}

ANTENNA Ky
{
datastreams = dsKy
phaseCalInt = 0
# April 2022 clocks
clockOffset = +1.1688 # n22hy01a 2022-04-06 offset plus 2022-04-21 delta
clockRate = -0.0000023938000000000000
clockEpoch = 2022y096d04h35m00s
}

SETUP defaultSetup
{
# tInt = 0.8192
# # subintNS = 819200000
tInt = 1.024
# specRes = 0.0625 # 16 MHz bw, 256 ch
specRes = 0.03125 # 32 Mhz / 1024 ch
doPolar = False
xmacLength = 0
strideLength = 0
# guardNS = 2000
# numBufferedFFTs = 20
}
```

```

RULE defaultRule
{
  setup = defaultSetup
}

# Note on DiFX v2d UDP_MTU :
#   DBBC3 : ethHeader:28B + psn64:8B      + VDIF:8224B : UDP_MTU = 8260
#   OCTAD : ethHeader:28B + psn64:8B(?) + VDIF:8224B : UDP_MTU = 8260

DATASTREAM dsKt
{
  networkPort = xxx
  UDP_MTU = 8260
  machine = n1
}

DATASTREAM dsKu
{
  networkPort = xzz
  UDP_MTU = 8260
  machine = n2
}

DATASTREAM dsKy
{
  networkPort = xxx
  UDP_MTU = 8260
  machine = n3
}

DATASTREAM dsKv
{
  networkPort = xxx
  UDP_MTU = 8260
  machine = n4
}

# EOPS from /cluster/difx/usno_finals.erp last updated at 2022-04-21 20:11:47 (+0000)
EOP 59689 { xPole=0.068120 yPole=0.462050 tai_utc=37 ut1_utc=-0.098893 }
EOP 59690 { xPole=0.070130 yPole=0.463090 tai_utc=37 ut1_utc=-0.098688 }
EOP 59691 { xPole=0.072060 yPole=0.463920 tai_utc=37 ut1_utc=-0.098303 }
EOP 59692 { xPole=0.073960 yPole=0.464670 tai_utc=37 ut1_utc=-0.097876 }
EOP 59693 { xPole=0.075820 yPole=0.465350 tai_utc=37 ut1_utc=-0.097515 }

```

다. 세종 + KVN 2Gbps e-VLBI 상관처리 프린지 검출

세종 belbi2 DiFX 서버를 이용한 세종+KVN 2Gbps e-VLBI 상관처리가 성공적으로 진행되었다. 세종과 KVN 각 망원경의 백엔드 시스템에서 2048 Mbps로 생성되는 데이터가 KREONET망을 통해 세종 우주측지관측센터의 belbi2 DiFX 서버로 손실 없이 전송되었으며, 6기 노드로 업그레이드 된 belbi2 서버는 2Gbps 4기의 e-VLBI 실시간 상관처리를 성공적으로 수행하였다. 데이터 전송속도는 ~2056Mbps로 안정적으로 유지되었으며, 웨이사 3C454.3 천체에 대해 성공적으로 프린지를 검출하였다. 검출된 프린지 신호의 신호대 잡음비(SNR)은 671이었다. 이상 1차년도에서 세종과 KVN 총 4기의 전파망원경으로 1Gbps e-VLBI 관측 성공에 이어, 세종 belbi2 DiFX 서버의 성능 향상 이후 2Gbps e-VLBI까지 성공적으로 상관처리 할 수 있음을 확인하였다.

```

Fri Apr 22 11:26:02 2022 0 belbi2 INFO Initialized
Fri Apr 22 11:26:02 2022 0 belbi2 INFO MPI Process 0 is running on host belbi2
Fri Apr 22 11:26:02 2022 0 belbi2 INFO About to process the input file..
Fri Apr 22 11:26:02 2022 0 belbi2 INFO DiFX VERSION = DiFX-2.5.2
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] datastream[0] had its array stride length automatically set to 32 based on nchan = 1024
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] datastream[1] had its array stride length automatically set to 32 based on nchan = 1024
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] datastream[2] had its array stride length automatically set to 32 based on nchan = 1024
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] datastream[3] had its array stride length automatically set to 32 based on nchan = 1024
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] had its xmac stride length automatically set to 128 based on numbers of output channels
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] had its rotate stride length automatically set to 16 based on xmacstridelen = 128
Fri Apr 22 11:26:02 2022 0 belbi2 DEBUG NS accumulate is 125 and max geom slip is 41,485, maxnsslip is 0
Fri Apr 22 11:26:02 2022 0 belbi2 DEBUG NS accumulate is 125 and max geom slip is 43,5639, maxnsslip is 0
Fri Apr 22 11:26:02 2022 0 belbi2 DEBUG NS accumulate is 125 and max geom slip is 40,8483, maxnsslip is 0
Fri Apr 22 11:26:02 2022 0 belbi2 DEBUG NS accumulate is 125 and max geom slip is 39,8873, maxnsslip is 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Config[0] had its guardns automatically set to 169 since it was not set in the configuration file
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Receive socket opened: socket is 17
Fri Apr 22 11:26:02 2022 0 belbi2 INFO STARTING mpifxcorr version 2.5
Fri Apr 22 11:26:02 2022 0 belbi2 STATUS Starting Version 2.5
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 0, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 1, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 2, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 3, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 4, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 5, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 6, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 7, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 8, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 9, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0

```

<그림 4-4> belbi2 서버에서 2Gbps e-VLBI 상관처리를 위한 DiFX 상관처리 초기화(화면캡처)

```

Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 27, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 28, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 29, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 30, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO For Visibility 31, offsetnsperintegration is 0, subintns is 64000000, and configindex is now 0
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Estimated memory usage by FXManager: 42,6038 MB
Fri Apr 22 11:26:02 2022 0 belbi2 INFO Hello World, I am the FxManager
Fri Apr 22 11:28:07 2022 0 belbi2 INFO Vis, 0 to write out time 0
Fri Apr 22 11:28:07 2022 0 belbi2 INFO 0/32 visibilities locked for accumulation, most recent index is 1
Fri Apr 22 11:28:07 2022 0 belbi2 INFO 1/32 visibilities ready to write out from 0
Fri Apr 22 11:28:07 2022 0 belbi2 INFO Min/Mean/Max number of subints processed is 0/0.666667/1, mincoreindex is 16, maxcoreindex is 0
Fri Apr 22 11:28:07 2022 0 belbi2 STATUS Running
Fri Apr 22 11:28:07 2022 0 belbi2 WEIGHTS 1,00 1,00 1,00 1,00
Fri Apr 22 11:28:07 2022 0 belbi2 INFO Vis, 0 is incrementing, since currentsubints = 16
Fri Apr 22 11:28:07 2022 0 belbi2 INFO The approximate mjd/seconds is 59691/8880
Fri Apr 22 11:28:08 2022 0 belbi2 INFO Vis, 1 to write out time 1,024
Fri Apr 22 11:28:08 2022 0 belbi2 INFO 0/32 visibilities locked for accumulation, most recent index is 2
Fri Apr 22 11:28:08 2022 0 belbi2 INFO 1/32 visibilities ready to write out from 1
Fri Apr 22 11:28:08 2022 0 belbi2 INFO Min/Mean/Max number of subints processed is 1/1.33333/2, mincoreindex is 8, maxcoreindex is 0
Fri Apr 22 11:28:08 2022 0 belbi2 STATUS Running
Fri Apr 22 11:28:08 2022 0 belbi2 WEIGHTS 1,00 1,00 1,00 1,00
Fri Apr 22 11:28:08 2022 0 belbi2 INFO Vis, 1 is incrementing, since currentsubints = 16
Fri Apr 22 11:28:08 2022 0 belbi2 INFO The approximate mjd/seconds is 59691/8881
Fri Apr 22 11:28:09 2022 0 belbi2 INFO Vis, 2 to write out time 2,048
Fri Apr 22 11:28:09 2022 0 belbi2 INFO 0/32 visibilities locked for accumulation, most recent index is 3
Fri Apr 22 11:28:09 2022 0 belbi2 INFO 1/32 visibilities ready to write out from 2
Fri Apr 22 11:28:09 2022 0 belbi2 INFO Min/Mean/Max number of subints processed is 2/2/2, mincoreindex is 0, maxcoreindex is 0

```

<그림 4-5> belbi2 서버에서 2Gbps e-VLBI 상관처리가 진행되며 상관처리된 visibility 데이터가 생성되는 모습(화면캡처)

```

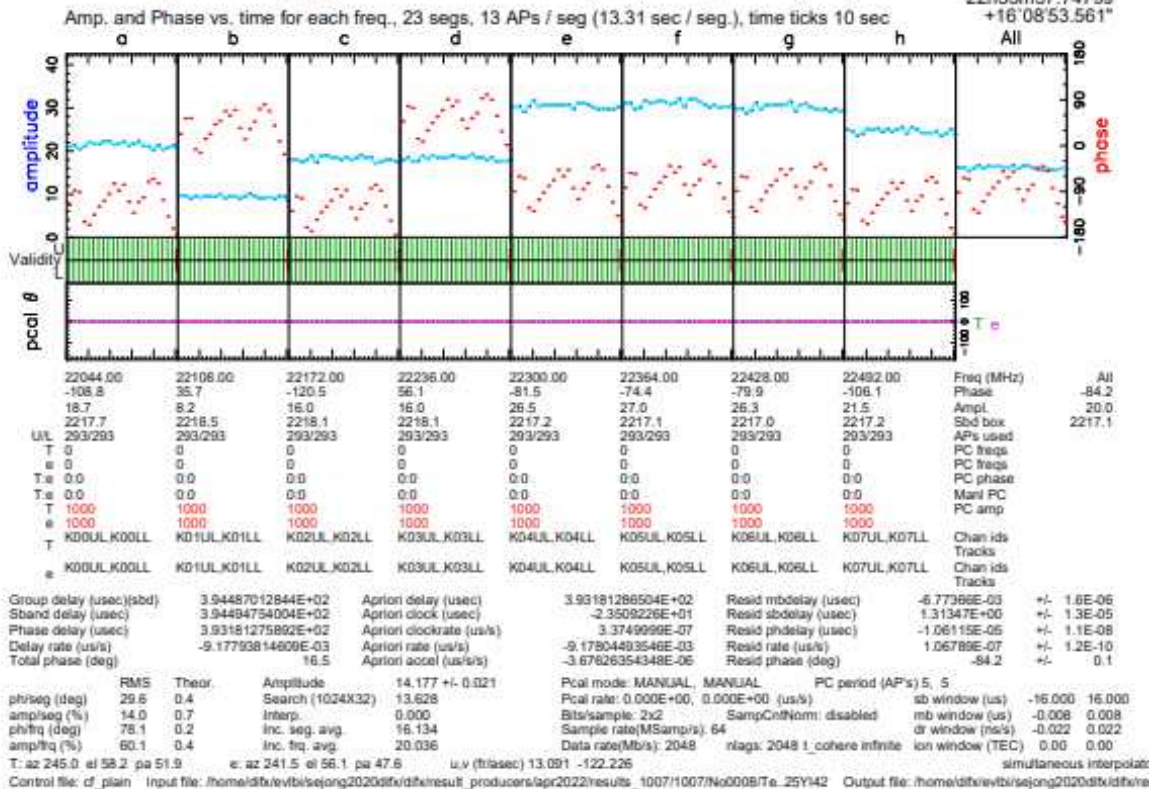
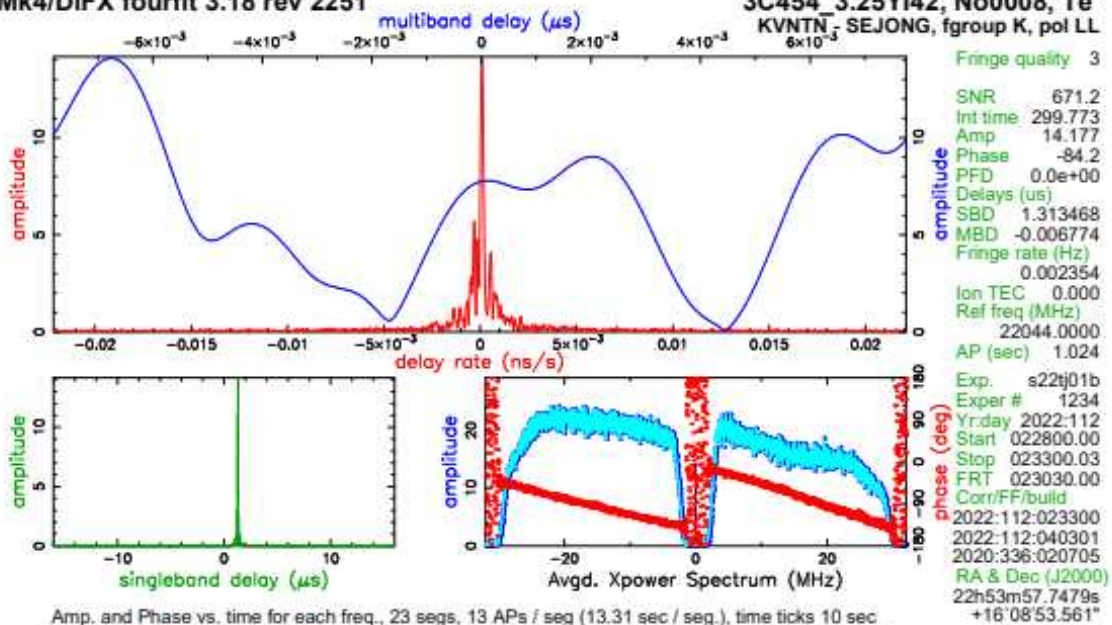
difaxn1 DiFX-2.5.2 -> errormon2
2022-04-22 10:35:44,659 DiFXAlert INFO MPI[ 1] n1.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2056.06 Mbps
2022-04-22 10:35:44,660 DiFXAlert INFO MPI[ 2] n2.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2056.05 Mbps
2022-04-22 10:35:44,660 DiFXAlert INFO MPI[ 3] n4.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2055.95 Mbps
2022-04-22 10:35:44,660 DiFXAlert INFO MPI[ 4] n3.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2056.03 Mbps
2022-04-22 10:35:46,298 DiFXAlert INFO MPI[ 3] n4.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2056.02 Mbps
2022-04-22 10:35:46,298 DiFXAlert INFO MPI[ 1] n1.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2055.89 Mbps
2022-04-22 10:35:46,298 DiFXAlert INFO MPI[ 2] n2.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2055.94 Mbps
2022-04-22 10:35:46,298 DiFXAlert INFO MPI[ 4] n3.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2055.87 Mbps
2022-04-22 10:35:47,936 DiFXAlert INFO MPI[ 3] n4.cluster s22tj01b_1003 Packets=51204 Dropped=0.00 Out-of-order=0.00
Rate=2056.11 Mbps

```

<그림 4-6> 각 망원경에서 belbi2 DiFX 서버로 2Gbps 데이터가 손실없이(dropped=0.0) 전송되는 모습(화면캡처)

Mk4/DiFX fourfit 3.18 rev 2251

3C454_3.25Y142, No0008, Te
KVNTN_3 SEJONG, fgroup K, pol LL



<그림 4-7> 세종+KVN 4기 전파망원경을 이용한 2Gbps e-VLBI DiFX 상관처리 프린지 검출

3. e-VLBI 운용을 위한 사용 매뉴얼 제작 및 e-VLBI 활용 방안

가. e-VLBI 운용을 위한 사용 매뉴얼

e-VLBI 운용을 위한 매뉴얼은 부록에 수록하였으며, 다음의 절차와 내용을 포함한다.

- 1. 세종 DiFX서버(belbi2) 설정**
 - 1.1 네트워크 인터페이스 설정
 - 1.2 SSH 사용자 설정
 - 1.3 e-VLBI를 위한 belbi2방화벽 설정
 - 1.4 OpenMPI를 위한 belbi2방화벽 설정
 - 1.5 DiFX멀티캐스트 메시지(Multicast Messages)에 대한 방화벽 설정
 - 1.6 관측국4기 x 2Gbps상과처리를위한 데이터 스트림 리다이렉션을위한 방화벽 설정
 - 1.7 관측국 4기 x 2 Gbps e-VLBI 상관처리
 - 1.8 SLURM설정
 - 1.9 DiFX설정
 - 1.10 Mark6기록기의 "MTU" 경로 설정 문제
- 2. 성능 측정**
- 3. 실시간 e-VLBI 상관처리**
 - 관측국3기 x 1024 Mbps 테스트
 - 관측국4기 x 1024 Mbps 테스트
 - 관측국3기 x 2048 Mbps 테스트
 - 관측국4기 x 2048 Mbps 테스트

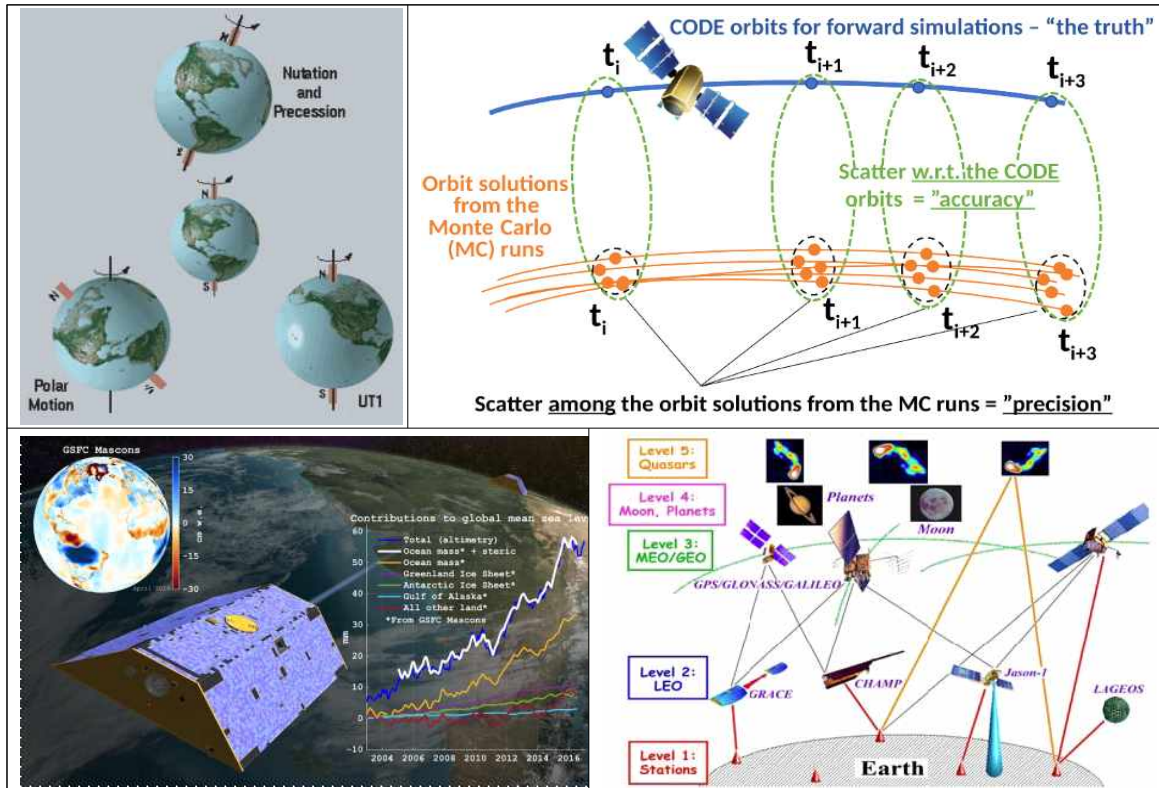
나. e-VLBI 활용 방안

e-VLBI의 가장 큰 차별성은 일반적인 VLBI 관측이 상관처리된 데이터를 얻기까지 수 주에서 수개월이 걸리는 것에 비하여, 상관처리가 실시간으로 진행되어 관측 결과를 바로 확인할 수 있다는 것이다. 이러한 빠른 상관관처리 결과 확인은 기존 VLBI 관측에 비해 ① 실시간 상관처리 결과 확인이 가능함에 따라 시스템 이상 여부 및 관측 성공 여부를 즉시 확인 가능하고, ② 관측 후 자료전송에 따른 업무량 감소 및 데이터 보관에 따른 비용을 절감할 수 있다. 이런 장점과 함께 e-VLBI 활용 방안은 다음과 같다.

③ 신속한 UT1 결정을 통한 응용 연구 분야 활용[3]

- 위성 및 우주탐사선의 정밀 궤도 결정(satellite tracking) [4]
- 우주탐사선 항법(Space Navigation)[5]
- 우주통신(space communication)
- 위성을 이용한 중력장 측정 (satellite gravity mission)
- 측지/천문 격변현상 연구[6]

④ e-VLBI를 활용한 차세대 인터넷망 테스트
 - 초대용량 데이터 전송 및 자료처리[7]



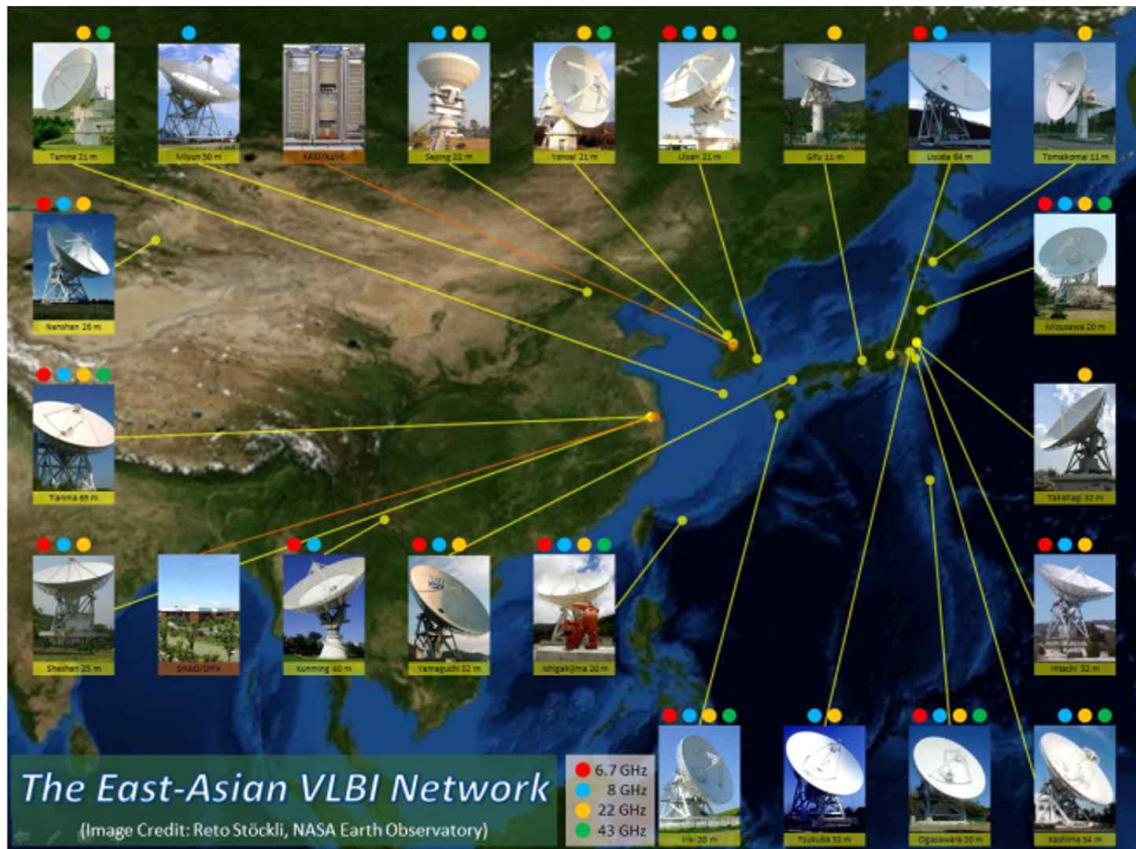
<그림 4-8> e-VLBI를 활용 분야. (좌상) 신속한 UT1 결정. (우상) 위성파 우주탐사선의 정밀 위치 결정 및 항법(Kloppetek et al. 2020). (좌하) 위성을 이용한 중력장 측정(NASA GRACE 미션 ©NASA). (우하) 우주측지 활용(GGOS ©IGG).

제5장

세종 VLBI 활용 확대를 위한 국·내외 VLBI 네트워크 협력 기반 구축

-
1. KVN·EAVN과 공동 관측·협력 방안 마련
 2. 국·내외 VLBI 관측망 참여를 위한 세종 VLBI 시스템 성능 측정 및
정량화된 성능지표 생성
 3. 세종+KVN·EAVN 등 VLBI 관측망을 이용한 활용 분야 발굴·제시

1. KVN·EVN과 공동 관측 협력 방안



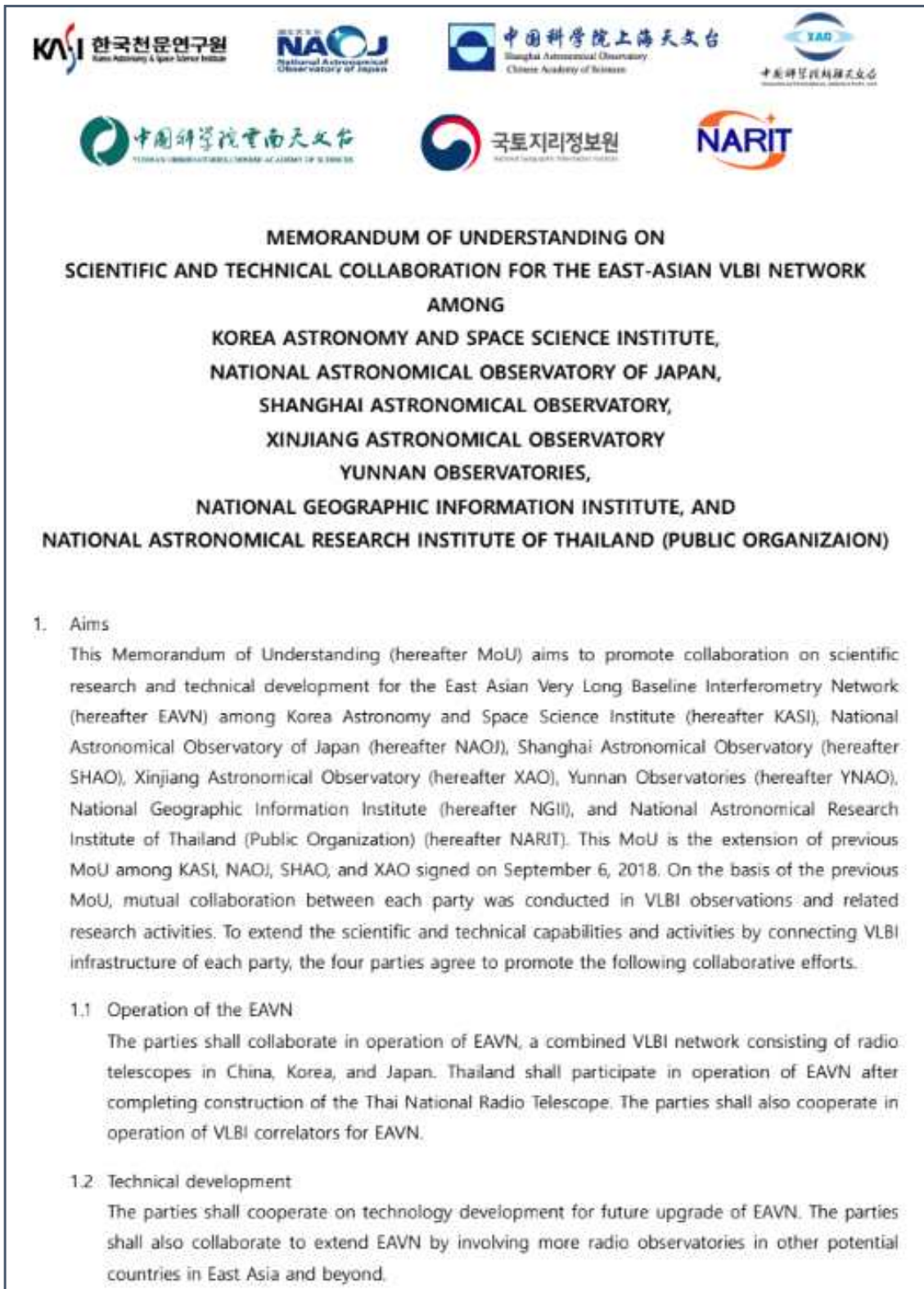
<그림 5-1> 동아시아 VLBI 망원경 분포도

한국우주전파관측망(KVN)은 서울, 울산, 제주에 직경 21m 전파망원경 3기를 하나의 네트워크로 연결하여 블랙홀 및 은하, 별의 생성과 사멸에 대한 높은 분해능 천체물리 관측연구를 진행하고 있으며, 특히 세계최초로 20 ~ 140GHz 사이 고주파수 대역 4곳을 동시에 관측할 수 있는 수신시스템을 독자 개발하여 2011년 교과부 ‘과학기술창의상’ 대통령상을 수상하였으며, 세계에서 가장 뛰어난 밀리미터(파장) 전파관측망으로 인정받고 있다. 세종 우주측지관측센터의 22m 전파망원경 또한 측지 Legacy VLBI 대역인 2/8GHz와 함께 22/43GHz 대역을 동시에 관측할 수 있는 매우 우수한 성능의 망원경으로 KVN과 공동 관측 연구 협력을 지속적으로 진행해오고 있다.

EAVN은 2005년 동아시아(한중일)에 분포하는 전파망원경들을 연결하여 직경 5,000km를 갖는 ‘동아시아 VLBI 네트워크(EAVN)’를 구성하기로 합의하고, 2013년부터 본격적인 성능측정 테스트를 진행한 것이 시초가 되었다. 이후 2018년 9월 한국천문연구원과 일본국립천문대, 그리고 중국의 상해천문대와 신장천문대 이상의 3개국 4개 기관을 중심으로 한 “동아시아 VLBI 관측망(EAVN: East Asian VLBI Network) 관측운영 및 과학기술협력에 관한 양해각서”가 체결되었다. 이로써 EAVN은 기존 유럽과 미국의 양대 VLBI 관측망인 EVN(European VLBI Network)과 VLBA(Very Long Baseline Array)와 함께 세계 3대 VLBI 관측망으로 자리매김하였다. 이 당시에는 세종이 공식적인 EAVN 멤버는 아니었지만, EAVN 측지 VLBI 관측을 다수 수행하며, KVN·EAVN과 긴밀한 공동 관측 및 과학기술 연구 개발 협력을 진행해오고 있다.

특히, EAVN 측지/측성 연구분야의 경우, 일본의 전파망원경들은 대륙판의 불안정으로 절대적인 위치 기준을 잡기가 매우 어렵고, 중국 망원경의 경우, 대륙판은 안정적이거나 상대적으로 망원경의 크기와 구동 속도의 제약으로 인해 VLBI 위치 기준으로서의 역할을 기대하기가 어렵다. 반면 세종은 IVS 관측에 매주 참여하며 절대위치를 잘 유지해오고 있으며, 상대적으로 안정적인 한반도에 위치해 있을 뿐만 아니라, 빠른 구동속도와 높은 망원경 감도 및 정밀도와 함께, 중국과 일본의 전파망원경 분포상 지리적으로도 중심에 위치해 측지 VLBI에서 상호 관측 가능한 천체의 개수 및 성과를 극대화시킬 수 있어 EAVN 측지/측성 VLBI 연구에서 중요한 기준 관측국의 역할을 수행할 수 있다. 실제로 세종이 EAVN 측지/측성 관측에 있어 보다 중추적인 역할을 수행해야 한다는 논의는 EAVN Galactic Astrometry 과학연구 그룹에서 다수 논의되었으며, 매달 개최되는 EAVN 운영회의(Regular Meeting) 및 분기마다 개최되는 EAVN 소장회의(Directors Meeting)에서도 논의 된 적이 있었다.

이러한 배경에서 2018년 EAVN 양해각서(MOU)가 갱신되는 시점인 2021년 10월에 국토지리정보원과 중국 윈난(Yunnan)천문대 그리고 태국 국립천문대(NARIT)의 EAVN 참여로 EAVN 관측운영 및 과학기술 협력에 관한 양해각서가 4개국 7개 기관으로 확대되어 체결되었다. 이러한 배경에서 이번 EAVN MOU 체결은 이미 상당수의 국제 IVS 측지 VLBI 망원경들이 측지/측성 및 천문분야 공동 활용을 통하여 많은 연구분야에서 시너지를 창출하고 있는 추세에 맞추어 세종 VLBI 망원경 활용을 확대하고, 이를 위한 국내외 VLBI 협력 네트워크 기반 구축하였다. 이후 세종은 매달 개최되는 EAVN 운영회의와 소장회의에 참여하며 KVN·EAVN과 긴밀한 공동 관측 협력을 유지하고 있다.



<그림 5-2> EAVN 공동운영 및 과학기술 협력에 관한 양해각서 체결(2021년 10월 1일). 한중일+태국의 4개국 7개 기관(국토지리정보원, 한국천문연구원, 일본국립천문대, 중국상해천문대, 중국신장천문대, 중국윤난천문대, 태국국립천문대)이 참여.

2. 국·내외 VLBI 관측망 참여를 위한 세종 VLBI 시스템 성능 측정 및 정량화된 성능지표 생성

VLBI 망원경의 활발한 활용을 위해서는 사용자 커뮤니티를 넓히고, 외부 사용률을 진작시키는 일이 필요하다. 이러한 일을 위해서 가장 우선적으로 준비해야하는 부분이 망원경의 성능지표를 산출하는 일이다. KVN과 EAVN의 경우, 각 망원경의 성능지표를 담은 망원경 성능 보고서(status report)를 홈페이지에 공개하고 있으며, KVN과 EAVN의 성능요약서는 다음에서 확인할 수 있다. 이번 과제를 통해 세종 22m VLBI 전파망원경의 성능을 측정할 수 있는 정량화된 지표를 생성하고 그 성능을 측정하여 ‘세종망원경 성능 보고서(status report)’를 제작하였으며, 부록에 수록하였으며, 이미 세종이 참여하고 있기 때문에 EAVN status report에도 세종 망원경에 관한 내용을 수록하였다.

□ KVN 망원경 성능 보고서: https://radio.kasi.re.kr/status_report/status_report.php?site=kvn

□ EAVN 망원경 성능 보고서 (세종 22m VLBI 전파망원경 내용 포함):

https://radio.kasi.re.kr/status_report/status_report.php?site=eavn

□ 세종 22m VLBI 망원경 성능 보고서 (부록 참조)

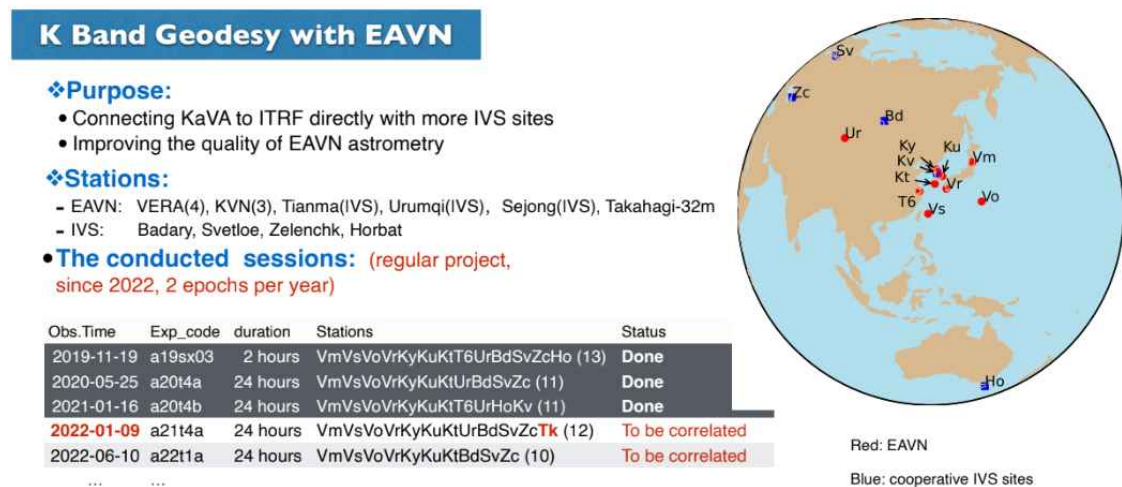
Status report 2022 of Sejong 22-m radio telescope			
1. Introduction			
이 문서에서는 국토정보지리원 세종 우주측지관측센터 22-m 전파망원경 (그림 1)의 K (22 GHz), Q (43 GHz) 밴드 대역 관측 성능 지표를 기술한다.			
세종 우주측지관측센터 22-m 전파망원경 (hereafter 세종 망원경)은 측지 연구를 주 목적으로 세종시에 건설된 22 m 직경의 전파 안테나로 S (2 GHz), X (8 GHz), K (22 GHz), Q (43 GHz) 대역을 관측할 수 있다. 이 중 S와 X 밴드 대역은 국제 VLBI 회원국 (IVS)과의 관측 수행을 목적으로 하고 있으며, K와 Q 밴드는 한국천문연구원의 밀리미터파 전용 VLBI인 KVN (Korean VLBI Network) 및 동아시아 VLBI인 EAVN (East Asia VLBI Network)과의 합동 관측을 위해 그 전파 수신기가 제작 및 설치되었다.			
따라서, K/Q band 대역 VLBI 관측을 위한 세종 전파망원경 성능에 대한 기준 지표가 필요하며, 여기에 그 성능 지표 측정 결과를 제시한다.			
우선, 현재 세종 전파망원경의 좌표는 표 1과 2와 같다.			
표 1. 세종 망원경의 geographic 좌표			
Antenna	Latitude (" ° ' ")	Longitude (" ° ' ")	Elevation (m)
KSJ (세종망원경)	127 18 11.0	36 31 22.0	156
표 2. 세종 망원경의 geocentric 좌표			
Antenna	X (m)	Y (m)	Z (m)
KSJ (세종망원경)	-3110079.9600	4082066.7340	3775076.8320
The position was measured in October 2014.			

<그림 5-3> 세종 22m VLBI 전파망원경 성능 보고서(status report)

3. 세종+KVN·EAVN 등 VLBI 관측망을 이용한 활용 분야 발굴·제시

가. EAVN의 측지·측성 VLBI 기준 역할 수행

세종이 EAVN에 본격적으로 합류하기 전 EAVN 측지 VLBI 관측 및 자료처리 분석은 일본을 중심으로 진행되었다. 특히, 일본 VERA의 Mizusawa 20m 전파망원경이 EAVN에서 수행하는 K-band(22GHz) 측지 VLBI 관측의 기준점으로 사용되었는데, 2011년 동일본 대지진 이후 사실상 절대 기준점을 잃어버린 상태로 진행되고 있었다. 그 결과 일본 VERA Mizusawa를 기준으로 산출한 KVN 및 VERA 전파망원경들의 절대좌표가 GNSS를 이용하여 측정한 망원경의 기준점(IVP)과 10cm 이상의 큰 차이를 보여왔다. 이후 중국 및 러시아, 그리고 호주의 VLBI 망원경과 연계하여 K-band 22GHz 측지 VLBI 관측을 수행하였고, 그 결과 KVN과 VERA의 절대좌표가 전체적으로 약 ~10cm 이상 북동쪽 방향으로 이동되어 있음이 밝혀졌다. 이는 절대 기준점을 잃은 VERA Mizusawa를 기준으로 측지 VLBI 데이터가 산출되어 발생한 것으로 사료되며, 이후 EAVN은 세종을 비롯하여 중국과 러시아, 호주를 포함하는 관측을 수행하고 있다. 앞 절에서도 설명했듯이, EAVN의 측지 VLBI에서 세종의 위치는 매우 중요하다. 세종은 기존 IVS 2/8GHz legacy 관측을 매주 수행하며 국제 측지 VLBI에서의 절대좌표를 잘 유지해오고 있을 뿐만 아니라, 22/43GHz 대역과 같은 고주파수 측지 VLBI 관측이 가능한 망원경으로 EAVN에서 진행하고 있는 20GHz 이상의 고주파수 측지 VLBI 관측의 안정적이면서도 가장 이상적인 기준점 역할을 수행 할 수 있다. 이미 세종은 이러한 역할을 수행하고 있으며, 이번 VLBI 데이터 분석 가속화 기술개발 연구를 통하여 광대역 고감도 관측 및 다양한 EAVN 관측 주파수를 지원할 수 있게 됨에 따라 앞으로 그 역할과 위상이 더욱 증대할 것이다.



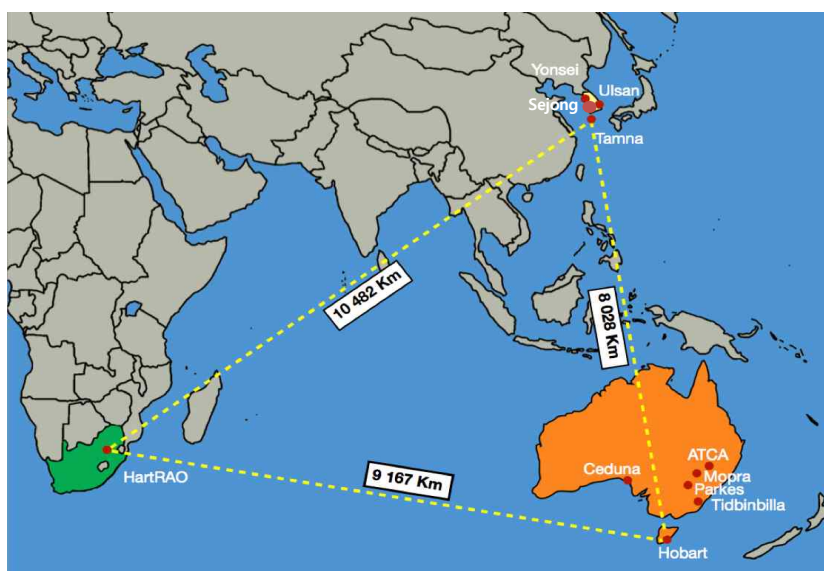
<그림 5-4> EAVN K-band(22GHz) 측지 VLBI 관측 현황(세종참여)

나. 국제 K-band 측지 VLBI 관측을 통한 천체기준좌표계(CRF) 연구

측지 VLBI 관측은 국제천체기준좌표계(ICRF: International Celestial Reference Frame)와 국제지구기준좌표계(ITRF: International Terrestrial Reference Frame)의 두 기준좌표계를 연결시키는 접점의 역할을 수행한다. 측지 VLBI 관측으로부터 측정하는 시간지연(delay)의 정밀도는 ICRF와 ITRF의 정밀도에 상호 의존한다. 따라서 두 기준좌표계의 정밀도 향상은 측지 VLBI에서 매우 중요한 요소이다.

기존 2/8GHz 측지 VLBI Legacy 관측 대비 고주파수 측지 VLBI의 가장 큰 장점은 바로 천체의 구조 영향으로 인한 오차 감소에 있다. 2018년 국제천문연맹 총회에서 발표된 ICRF3는 처음으로 기존 S/X-band(2/8GHz) 천체 4,536개 이외에도 K-band(24GHz) 824개 천체와 Ka-band(32GHz) 678개 천체를 포함하였다. 앞으로도 고정밀 기준계 실현과 VGOS에서 목표하고 있는 VLBI 관측으로부터 1mm 위치 정밀도를 달성하기 위해서는 고주파수 측지 VLBI 관측이 필수적이다.

특히, 천체기준좌표계(CRF)의 정밀도 향상을 위해서는 현재 북반구에 치우쳐져 있는 K-band 천체들의 개수를 현저하게 늘리는 것이 매우 중요하다. 따라서 세종과 KVN, 남아프리카공화국의 HartRAO, 호주의 Hobart 전파망원경이 구현할 수 있는 8,000km 이상, 최대 10,000km에 이르는 동서-남북 방향의 초장기선에서 K-band 천체들을 관측하는 제안서를 제출하였다. 이 관측제안서는 미해군천문대, NASA, 호주, 남아프리카 공화국의 연구자들과 국토지리정보원 세종 우주측지관측센터의 연구자들이 함께 참여하여, K-band에서 707개 천체를 관측하여 남북 방향의 CRF 천체 개수를 늘리고, 기준좌표계 정밀도를 향상시키고자 한다.



<그림 5-5> 천체기준좌표계 성능 향상을 위한 세종+KVN+남아프리카공화국+호주의 전파망원경 분포

KOREAN VLBI NETWORK OBSERVING APPLICATION

VLBI	Proposal ID: 2023A-00	Received Date: 2022/ /
TERM: 2023A		

1. Title of proposal: Improving the K-band CRF with North-South Baselines

2. Antenna: (Fill on the last line)

Name	E-mail	Institution/Country	Student
Aletha de Wit	aletha@sarao.ac.za	SARAO, South Africa	No
Taeheun Jung	tjung@kasi.ac.kr	KASI, Korea	No
David Gordon	david.gordon@usno.navy.mil	USNO, United States	No
Christopher Jacobs	chris.jacobs@nasa.gov	NASA/JPL, United States	No
Jong-Ar Lee	lee@kasi.ac.kr	KASI, Korea	No
Janie McCallum	janie@utas.edu.au	UTAS, Australia	No
Hong Jong Oh	hong@ngii.ac.kr	NGII, Korea	No
Shuangling Xu	xu@kasi.ac.kr	KASI, Korea	No
Hoon Yoon	hyoon@ngii.ac.kr	NGII, Korea	No

3. Contact author: Name: Aletha de Wit E-mail: aletha@sarao.ac.za Phone: +27-22-754-2085 FAX: +XX-XXX-XXX-XXXX

4. Staff support:
 - Observing setup: ☐ None ☐ Consultation ☐ Extensive help
 - Post processing: ☐ None ☐ Consultation ☐ Extensive help

5. Proposal type: ☐ KVN-Sejong (Shared risk mode) ☐ Joint proposal (If joint, network name:)

6. Scientific categories:
☐ Galactic ☐ Extragalactic ☐ Astrometry ☐ Geodesy ☐ Radio transient and pulsars
☐ AGN ☐ Matter ☐ Galactic center ☐ Star formation ☐ Evolved star

7. Observing type:
☐ Continuum ☐ Spectral line ☐ Phase referencing ☐ Polarimetry
☐ Survey ☐ Multi-frequency ☐ Target of opportunity

8. Observing frequency and polarization:
☐ 22GHz ☐ 86GHz ☐ 129GHz
☐ Single polarization ☐ Dual polarization ☐ State that Sejong is available at 22/43 GHz (1/20bps) only.

9. Observing sessions: ☐ single epoch ☐ multiple epochs
 - Total time requested: hrs
 - Number of sessions: Number of hour each: hrs Separation: hrs
 - Max/Min LST (J2000.0): -
 - Preferred range of dates or dates which are NOT acceptable:

10. Abstract (200 words max, 10 point)
 We propose K-band VLBI observations of 307 sources to improve coverage with north-south baselines between Korea, Australia and South Africa in order to refine the K-band component of the current standard International Celestial Reference Frame (ICRF-3). The K-band component of the ICRF-3 is about a factor of ten less precise in the declination than in right ascension due to a lack of southern stations and long north-south baselines. The Korean VLBI Network (KVN) and Sejong in Korea, Australia would fill this need with its north-south baseline component of > 9000 km, being 2 times the length of the current longest north-south baseline in the K-band ICRF. In addition, Hartbeesthoek, South Africa adds baseline components > 9000 km that will improve the accuracy of source positions in both coordinates. Thus data from the proposed work has potential to significantly improve the precision of the K-band ICRF. As the ICRF defines the official IAU system of angular coordinates on the sky, the proposed work would benefit many disciplines including global and differential astrometry, geodesy, earth orientation, and geophysical seismology. In particular, these observations will be the KVN and Sejong in the global VLBI K-band astrometric network.

When your proposal is scheduled the contents of this application form (but not supporting materials) will be made public. This PDF form was generated on October 28, 2022

Improving the K-band Celestial Reference Frame with North-South Baselines

P.I: Aletha de Wit¹ (SARAO, South Africa)
 Co-I: Taehyun Jung (KASI, Korea), David Gordon (USNO, US),
 Christopher Jacobs (JPL, Caltech/NASA, US), Jong-Ar Lee and Shuangling Xu (KASI, Korea),
 Janie McCallum, (UTAS, Australia), Hoon Yoon and Hong Jong Oh (NGII, Korea)

K-band astrometric observations between Korea, Australia and South Africa would benefit both the International Celestial Reference Frame (ICRF) and the Korean VLBI Network (KVN) user communities. Not only will these observations diversify and strengthen the K-band Celestial Reference Frame (CRF) network geometry and improve the astrometric accuracy, but it would be Korea to the global VLBI K-band astrometric network and ensure that the KVN becomes a contributor to the next generation ICRF. It would also allow for more accurate KVN station positions and would provide a list of well observed and well studied K-band calibrator sources accessible to the KVN, extending the KVN 22-GHz calibrator catalog from a few 100 km intra-Korea baselines to 7000-9000 km baselines. In addition, these observations would complement and extend the ongoing K-band geodesy campaign on the East Asian VLBI Network (EAVN).

1 Scientific context

1.1 High-Frequency Celestial Reference Frames

In the last few years, considerable work has been done and significant progress made on defining celestial reference frames (CRFs) at higher radio frequencies, such as K-band (21 GHz) [1] and Ka-band (32 GHz) [2]. The advantage of observing at K-band, is that many radio observatories typically have K-band receivers, while Ka-band receivers are typically only available at spacecraft tracking stations which are very few in number. The advantages of K-band over the historically standard S/X-band (2-8.4 GHz) frequencies are (1) more compact structure and (2) ~3 times higher resolution (see Figure 1). On VLBI scales at higher radio frequencies, extragalactic radio sources tend to exhibit more compact source structure and a reduced effect of core-shift, e.g., [3, 4]. For these reasons, astrometric VLBI observations of reference sources at higher radio frequencies permit the construction of a more accurate and more stable reference frame, which will be advantageous in tying the VLBI frame to optical frames such as Gaia, e.g., [5, 6]. In addition to the more compact source morphology, higher radio frequencies also allow observations closer to the Sun and the Galactic Plane. Differential VLBI astrometry on water masers to improve our understanding of the Galactic structure (e.g. the BeSSeL project [7]) is only possible at K-band, and thus requires well-understood K-band calibrator sources.

1.2 Construction of a Celestial Reference Frame at K-band

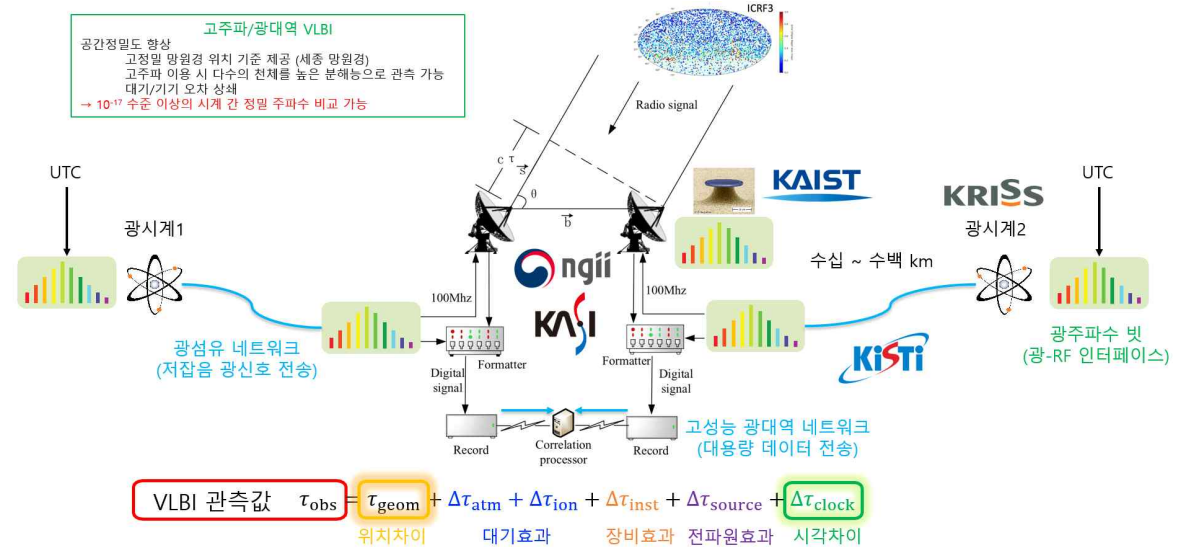
Astrometry [8] and imaging observations [4] led by Lauer 2002-10 provided a foundation for the development of a reference frame at K-band consisting of 269 sources. However, this work lacked coverage below -45°, had several large empty regions north of -45°, and had uncertainties in source positions at the 100 μ as level. In 2014, a new K-band collaboration was formed in order to produce full sky coverage, improve spatial density, and improve astrometric accuracy, e.g., [1]. In August 2018, during the XXXth General Assembly of the IAU, the K-band CRF was adopted as part of the third generation International Celestial Reference Frame (ICRF-3, [9]). The ICRF-3 contains source positions at X-band, as well as independently estimated source positions at K- and Ka-band. Since then, there have been many efforts to continue the improvement of the ICRF, including the high-frequency frames, thereby answering the IAU's call for maintenance of the ICRF-3 (IAU Resolution B2, 2018*).

The current K-band CRF consists of 1035 relatively uniformly distributed sources—comparable to the number of regularly observed S/X sources. It is constructed from 1.76 million observations collected in 122 observing sessions [10]. These observations are comprised of ongoing sessions using the Very Long Baseline Array (VLBA) in the United States and single-baseline sessions between the Hartbeesthoek

*Vice-President of IAU Commission A1 Astrometry, Chair of IVS CRF Committee, Member of ICRF-4 Working Group
https://www.iau.org/static/resolutions/IAU2018/ResB2_English.pdf

<그림 5-6> 세종+KVN+남아프리카공화국+호주 전파망원경을 이용한 K-band 측지 VLBI 관측 제안서(제출본)

다. 초정밀 시공간 융합연구



<그림 5-7> 대륙간 광시계 비교를 위한 시공간 융합연구 개념도

2019년 물리상수를 기반으로 한 국제단위계(SI)의 재정의 이후, 가장 높은 정확도를 지니고 있는 시간 단위인 ‘초’도 2030년경 재정의의 앞두고 있다. 현재의 세슘원자의 마이크로파 주파수보다 1만 배 높은 광주파수 시계 기반으로 바뀌게 될 예정이다. 이를 위해서는 전 세계 각국에서 개발된 광주파수 시계의 동등성 확보를 위해 10^{-17} 이상의 정밀도로 시각/주파수를

대륙 간에 비교할 수 있는 기술력 확보가 시급하다.

한국표준과학연구원(KRISS)의 광주파수시계(이하 광시계)와 수소원자시계, 국가과학기술정보연구원(KISTI)의 국가과학기술연구망(KREONET), 한국천문연구원(KASI)이 보유한 한국우주전파관측망(KVN)과 국토지리정보원이 보유한 세종 우주측지관측센터의 전파망원경을 통해, 고주파수 광대역 VLBI 관측을 수행하여, 대륙간 시각/주파수 비교를 통한 세계 시간단위(“초(second)”)의 재정의에 기여하고 지구/천체 기준좌표계의 정밀도 한계를 극복하기 위한 융합연구를 제안하였다.

국제단위계(SI) 가운데 전류(A), 온도(K), 질량(kg), 물질량(mol)은 2019년 재정의가 이루어진 반면, 가장 높은 정확도를 지니고 있는 시간 단위인 ‘초’는 기술적 한계로 인해 1967년 정의된 이후 반세기가 지나도록 재정의 되지 못하고 있는 실정이다. 현재 국제단위계(SI)의 시간 측정 단위인 ‘초’는 전 세계 80여 개 기관이 보유한 수백 대의 상용 원자시계와 세슘 원자시계로부터 생성되는 세슘원자(Cesium)의 고유 주파수 측정 결과를 인공위성을 이용해 상호 비교하는 방식으로 시간을 관리하고 있다. 기존의 세슘원자시계보다 약 100배 정확한 광시계가 개발됨에 따라 ‘초(second)’재정의 관련 연구는 전 세계 과학계의 초미의 관심사다.

이를 위해서는 세계 각 기관에서 개발한 광시계 주파수간 비교가 필요하며, 대륙간에 떨어져 있는 광시계 시각 정보를 광대역 VLBI를 활용해 상호 비교할 수 있는 방법을 제안하였다. 광시계 비교를 위한 광대역 VLBI 관측은 광시계 신호로부터 잡음 없는 고주파 신호 생성 기술을 필요로 하며, 대용량의 데이터를 높은 안정도로 빠르게 전송할 수 있는 전송망 인프라 역시 필수적이다. 세종과 KVN으로 구성된 한반도 VLBI 관측망은 고주파수(10GHz 이상) 대역에서 다(多)주파수(22/43/(86/129GHz) 동시 관측이 가능한 독창적 시스템으로, 다주파수 광대역(최대 ~100GHz 대역) VLBI 관측을 통해 공간분해능과 시간지연 측정 정밀도를 기존 대비 10배 이상 향상시킬 수 있고, 한반도 전역의 전파망원경이 KREONET을 통해 시동기를 할 수 있는 최적의 인프라를 갖추고 있다.

이에 우리나라에서 해당 분야 연구에 주도적으로 참여하고 있는 국내기관 간 융합연구 협력을 추진하였다. 한국과학기술정보연구원이 운영하고있는 국가과학기술연구망(KREONET)을 기반으로 한국표준과학연구원이 개발한 이터븀(Ytterbium) 광시계, 한국과학기술원 광주파수빗(optical frequency comb) 기술, 한국천문연구원의 한국우주전파관측망(KVN)과 국토지리정보원 우주측지관측센터 전파망원경이 결합된 한반도 VLBI 관측망을 활용해 세계 최초 시공간 측정 정밀도 한계를 극복하기 위한 융합연구를 수행하기 위하여 2021년 11월 5개 기관장이 모인 업무협약 체결식을 진행하였고, 2022년 6월 국가과학기술연구회 창의형 융합연구 사업에 선정되어 진행 중이다. 본 VLBI 데이터 분석 가속화 기술개발 연구를 통하여 이러한 초정밀 시공간 융합연구에 매우 중요한 기반 인프라와 필요 성능을

제공할 것으로 사료된다.

□ 시공간 융합연구 소개 유튜브 링크 (국토부 제작): https://youtu.be/uH4Ms_I1Noc

- 광대역 고주파수 측지 VLBI기반 대륙간 광시계비교를 통해‘초(Second)’의 재정의에 기여



<그림 5-8> (좌) 시공간 융합연구를 위한 5개 기관 업무협약 체결식 (2021년 11월 24일).
(우) 체결된 시공간 융합연구 양해각서

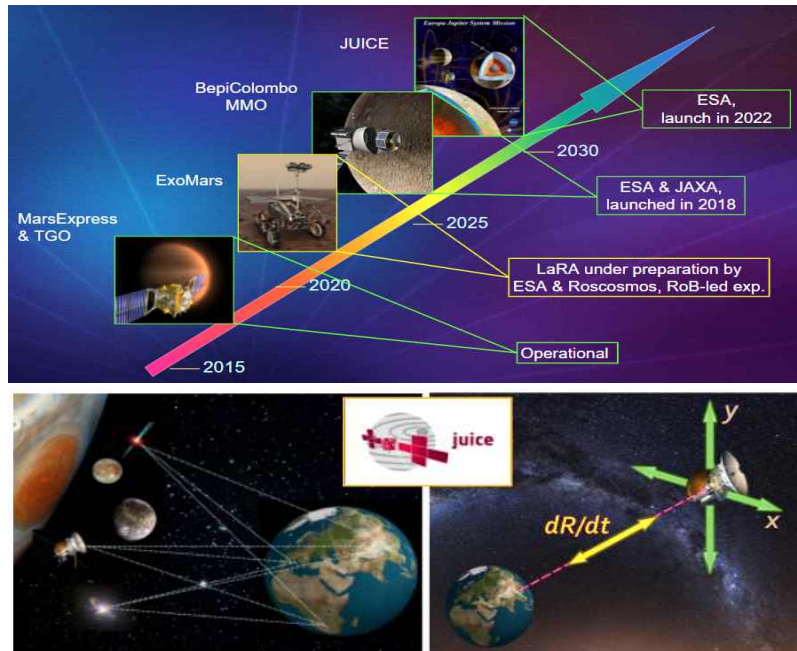
라. 국·내외 VLBI 전파망원경을 활용한 우주탐사선 정밀 위치 결정

VLBI 기술을 활용하여 유럽이나 미국, 중국에서는 오랫동안 우주탐사선의 항법 및 정밀 궤도를 실시간으로 산출하는 원천 기술을 선도해 왔다. 대표적으로 유럽 EVN(European VLBI Network), 미국 DSN(Deep Space Network), 그리고 중국 CVN(Chinese VLBI Network) 등과 같은 VLBI 관측망에서는 위성이나 우주탐사선과의 통신에 사용되는 전파신호를 이용하여 실시간으로 관측하고 상관처리하는 e-VLBI 기술을 활용하여 우주개발의 원천기술에 해당되는 정밀 위치(궤도)를 결정하고 있다.

이러한 VLBI 추적(Tracking) 기술은 기존 레인지(range)나 도플러(doppler) 추적 기술과 병행함으로써 우주탐사선의 3차원 공간상의 위치를 가장 높은 정밀도로 결정할 수 있다. 미국의 보이저호를 비롯하여, 유럽우주국(ESA)의 화성탐사선이나 금성탐사선, 중국 창어3 & 4와 같은 사례들이 있다. KVN은 ‘12년 EAN과 함께 X-band(8GHz)에서 화성탐사선의 위치를 추적하는 VLBI 관측에 참여한 바가 있다.

ESA는 2023년 4월에 발사하여 2031년 7월에 목성에 도달하는 우주탐사선 JUICE(Jupiter Icy Moons Explorer)를 발사하여 목성의 위성들에 관한 다양한 과학임무를 수행 할 예정이다. 이 미션의 공식명칭은 ‘PRIDE (Planetary Radio Interferometry and Doppler Experiment)’로 EVN의 주관기관인 네덜란드 JIVE에서 해당 과학임무를 수행한다. JIVE는 2005년 호이겐스(Huygens) 탐사선 VLBI 추적 실험의 후속 프로그램인 JUICE 우주

탐사선의 위치를 정밀하게 추적하여 목성과 얼음 위성의 중력장을 측정하는 연구 목표를 갖고 있다. ESA PRIDE-JUICE 프로그램을 수행하는 JIVE 책임자로부터 세종과 KVN의 전과 망원경의 참여를 요청받았으며, 관련 논의를 진행하고 있다.



<그림 5-9> (상) 유럽우주국(ESA)의 우주탐사 미션
(하) ESA JUICE 목성탐사선의 VLBI 관측 개념도 (© ESA/JIVE)

세종+KVN과 더불어 항공우주연구원에서 운영하고 있는 지상국 안테나도 VLBI 백엔드 시스템을 갖춘다면 우주탐사선 추적에 활용할 수 있으며, 관련 협력을 위한 논의가 진행 중이다.

Radio Antennas in S. Korea

KVN Yonsei (KASI)

- 21m
- K/Q/W/D
- + (C/X/Ka) planned

Geodetic Antenna (NGII)

- 22m
- S/X, K/Q

KVN Tamna (KASI)

- 21m
- K/Q/W/D
- + (C/X/Ka) planned



KVN PyeongChang (KASI)

- under construction
- 21m
- K/Q/W/D + 170/230GHz
- + (C/X/Ka) planned



Korean Deep Space Antenna (KARI)

- under construction
- 35m
- S/X

KVN Ulsan (KASI)

- 21m
- K/Q/W/D
- + C/X (room temp. Rx)

<그림 5-10> 우리나라의 VLBI를 활용한 우주탐사선 추적 가능 인프라(가능성)

VLBI 데이터 분석 가속화 기술개발 연구[2차]

제6장

광대역 e-VLBI 관측을 위한 주파수 합성기 개발 및 적용·운영방안 마련

1. 세종 시스템에 적합한 합성기 개발을 위한 시스템 구성도 분석·평가
2. 주파수 합성기 개발을 위한 요구 성능 설계
3. 주파수 합성기 회로 설계 및 제작
4. 주파수 합성기 종합성능 테스트
5. 세종 VLBI 시스템 적용을 위한 현장 성능 시험
6. 22/43GHz 광대역화 구현을 위한 주파수 합성기 적용 방안 마련

광대역 e-VLBI 관측을 위해 기존의 주파수 대역보다 넓은 주파수 대역을 관측하기 위해서 국부발진기로 사용될 주파수 합성기를 개발하고자 한다. 기존 22/43GHz 수신기는 13.25/17.055GHz 의 단일 주파수의 국부발진기를 사용하기 때문에 각각 2GHz의 대역폭인 21.25-23.25GHz 대역과 42.11 - 44.11GHz 대역의 관측이 가능하다. 현재의 관측 주파수 이외의 주파수 대역을 관측하기 위해서는 주파수 합성기를 사용하여 국부 발진 주파수를 가변하여 더 넓은 주파수 대역을 관측할 수 있다.

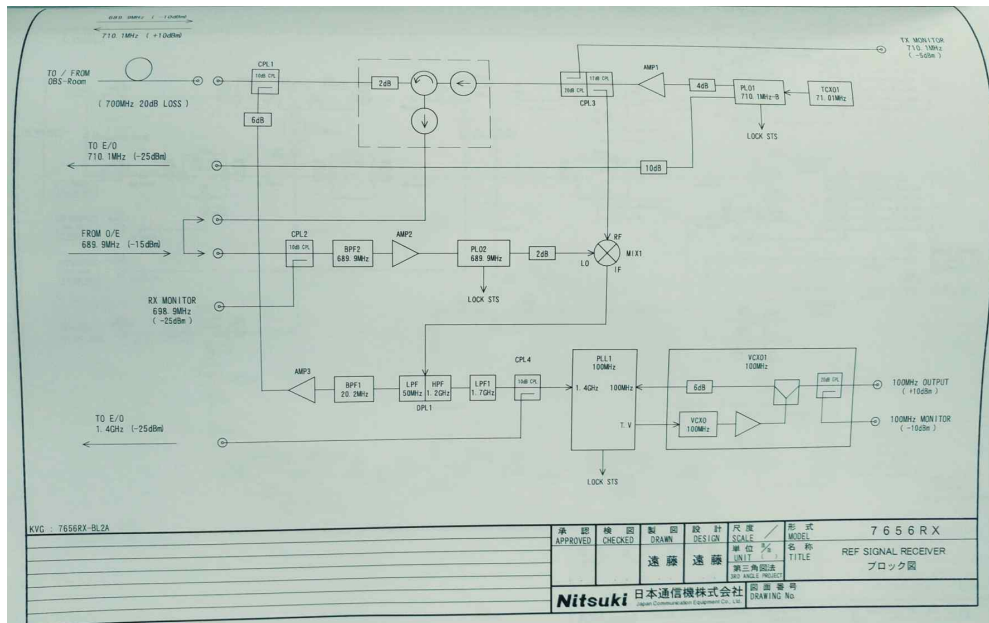
본 과제에서는 현재 수신 시스템을 분석하여, 더 넓은 주파수 대역 관측이 가능한 수신기에 필요한 주파수 합성기 규격을 제안하고, 이를 개발, 운영하기 위한 방안을 제시하려고 한다.

1. 세종 시스템에 적합한 합성기 개발을 위한 시스템 구성도 분석, 평가

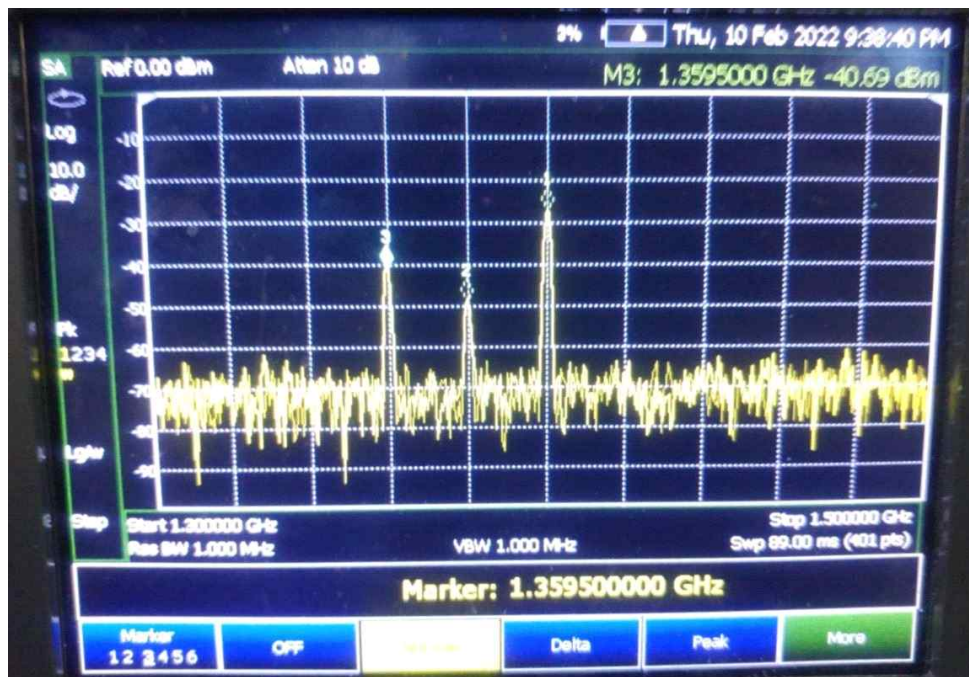
가. 기준신호 전송 시스템 분석, 평가

세종 VLBI 망원경 수신기를 위한 광대역 주파수 합성기의 기준 주파수 공급을 위해 기존의 기준신호 전송시스템을 분석하였다. <그림 6-1>에 안테나 수신기 실에 설치된 기준신호 전송시스템의 블록도를 보였다. 세종 VLBI 망원경 기준신호 전송시스템은 689.9MHz와 710.1MHz 신호를 사용하여 H-maser 1.4GHz에 동기화 되는 1.4GHz 신호를 안테나 수신기실에서 만들고, 이로부터 100MHz 와 10MHz 기준 주파수를 만들어 수신기 국부발진기에 공급한다.

1400MHz에서 100MHz를 만드는 과정에서 수신기 실 내부 온도 변화의 영향을 받기 때문에, 1400MHz를 직접 사용하여 주파수 합성기의 주파수를 동기화하는 것이 수신기 위상 안정도에 유리하다. 기준신호 전송시스템에서 1400MHz 신호 출력이 있는지 확인하였다. 기준신호 전송시스템의 후면에는 1.4GHz를 H-maser가 설치된 관측기기실에 전송하기 위한 포트가 설치되어 있다. 이 포트의 출력 신호를 측정하여 그림 6-2에 보였다.



<그림 6-1> 기준신호 전송시스템의 안테나 수신기 실 블록도.
(참조: 세종 VLBI 망원경 기준신호 전송시스템 매뉴얼)



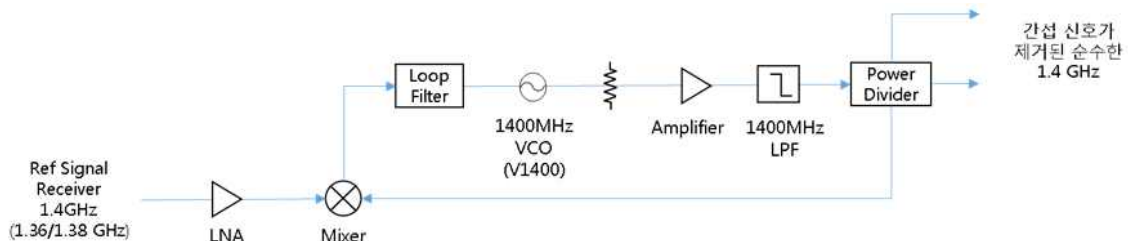
<그림 6-2> 기준신호 전송시스템 1.4GHz 포트의 출력 스펙트럼. 1.4GHz 신호의 크기는 -25 dBm이고, 1.36GHz, 1.38GHz 신호가 같이 출력된다.

그림 6-2에서 1.4GHz에서 -25 dBm의 출력 전력을 보였다. 이는 그림 6-1의 블록도의 값과 같다. 하지만 1.4GHz 신호 외에 1.36GHz 와 1.38GHz 신호도 발견되었다. 1.4GHz 신호를 주파수 합성기의 기준 주파수를 사용하기 위해서는 1.36GHz와 1.38GHz 신호를 제거하고 1.4GHz 신호만 남겨야 한다. 제거할 주파수 신호가 1.4GHz 신호에 매우 근접하여 필터로는 제거가 어려워 PLL(Phase Locked Loop)를 사용하여 제거해야 한다.

현재의 기준신호 전송시스템에서 1GHz 이상의 기준 주파수 공급이 가능하다는 것이 확인하였고, 추가로 1.36/1.38GHz 의 인접 신호 제거를 위해 PLL 회로를 설치하면 안정된 높은 기준 주파수 신호 공급이 가능함을 확인하였다.

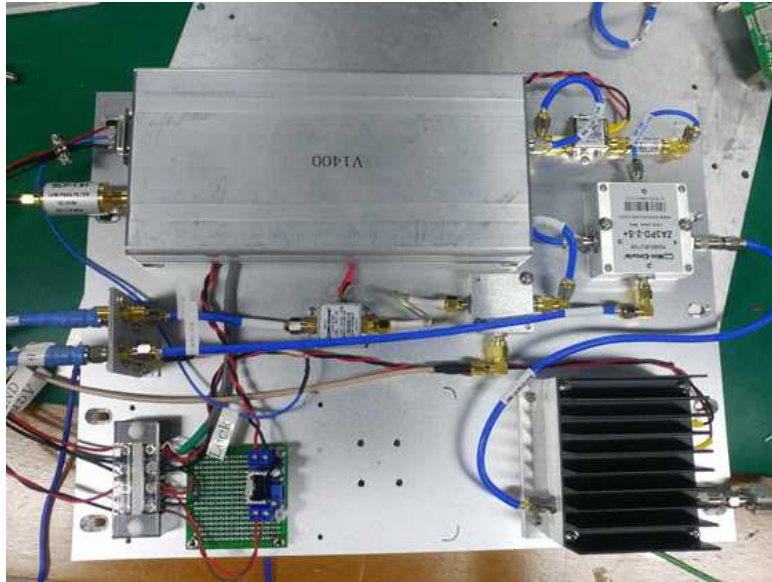
나. 10/100MHz 이외의 1GHz 이상의 고주파수의 기준 주파수 공급 방법 검토

그림 6-1과 6-2에서 1.4GHz 신호를 사용 가능함을 확인하였다. 1.4GHz 기준 주파수와 인접한 1.36/1.38GHz 신호 제거를 위해 그림 6-3과 같이 PLL 회로를 구성하여 1.4GHz 기준 주파수만 공급할 수 있다. 이 PLL 회로에서 1400MHz VCO(Voltage Controlled Oscillator)는 주파수 조정 범위가 수 kHz 정도로 좁아서 1.36GHz나 1.38GHz 에는 동기가 되지 않고 1.4GHz에만 동기가 될 수 있어야 한다. 이 회로를 통해 순수한 1.4GHz 신호를 만들 수 있는 PLO(Phase Locked Oscillator)를 구성하였다.

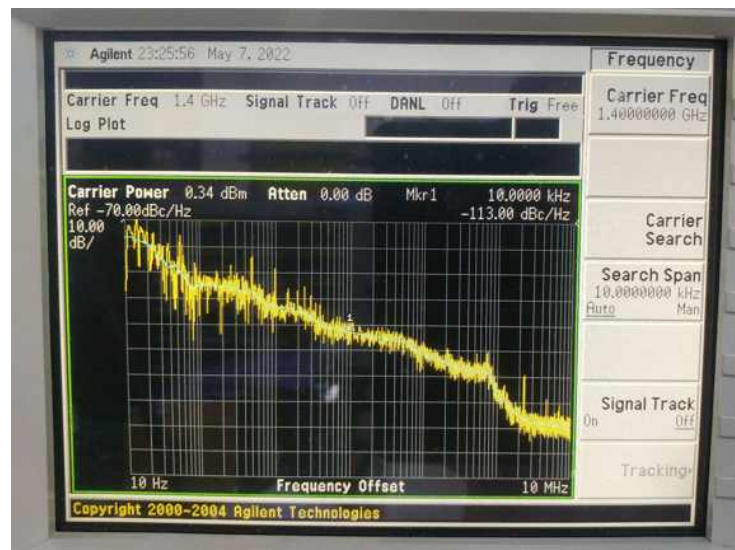
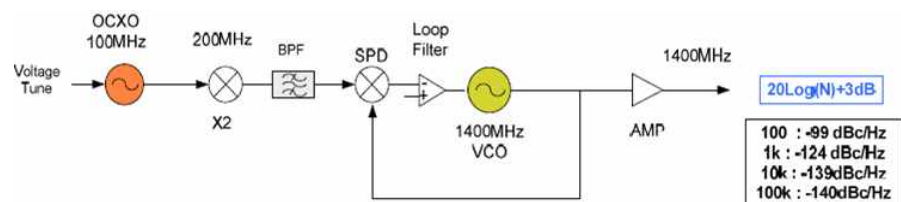


<그림 6-3> 1.36/1.38GHz 간섭 신호 제거를 위한 1.4GHz PLL 회로; 1400MHz VCO의 tuning range는 수 kHz 정도임.

그림 6-4에 조립된 1.4GHz PLL 회로를 보였다. 그림 5(상)의 블록도를 가진 1400MHz VCO를 활용하여 PLL 회로를 구성하였다. 구성된 1.4GHz PLL 회로의 출력 신호의 위상 잡음을 그림 6-5(하)에 보였다. 1.36GHz와 1.38GHz 신호가 제거된 스펙트럼은 별도로 보이지 않았다.



<그림 6-4> 1.36/1.38GHz 간섭신호가 제거된 1.4GHz 신호 출력을 위한 PLL 회로 제작

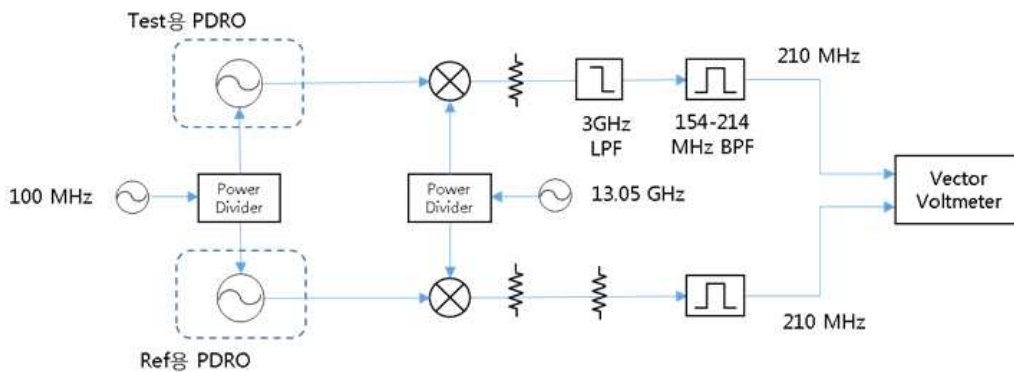


<그림 6-5> (상) 1.4GHz PLL 회로에 사용된 V1400, 1.4GHz VCO의 구성도.
OCXO(Morion,MV218)의 주파수 조정 범위의 제한을 받아서 +/- 4.2KHz의 주파수 조정 범위를 갖는다.

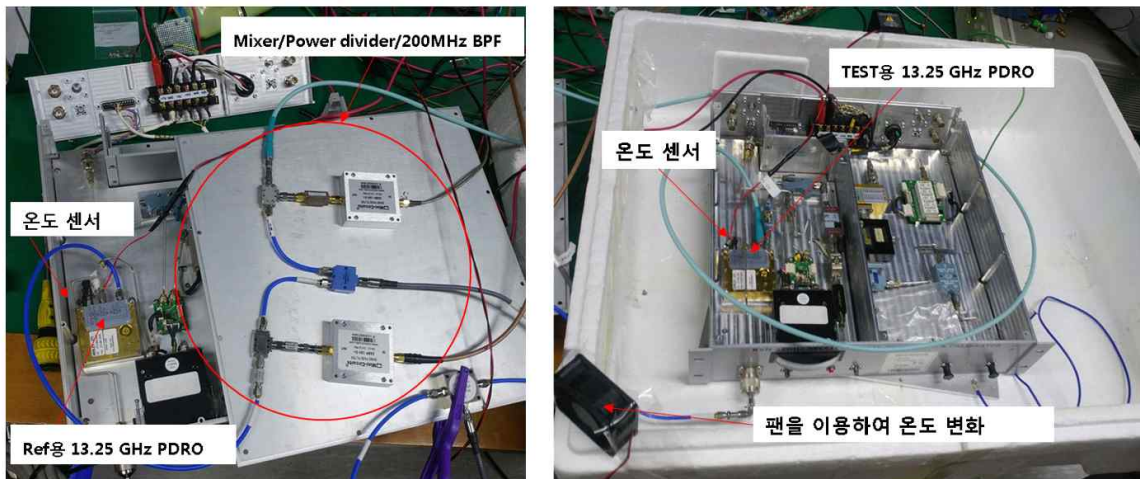
(하) 1.4GHz PLL 회로의 출력 신호의 위상 잡음 측정 결과

다. 현재 설치된 22/43GHz 수신기 및 주파수 하향 변환기의 주파수 안정도 분석, 평가

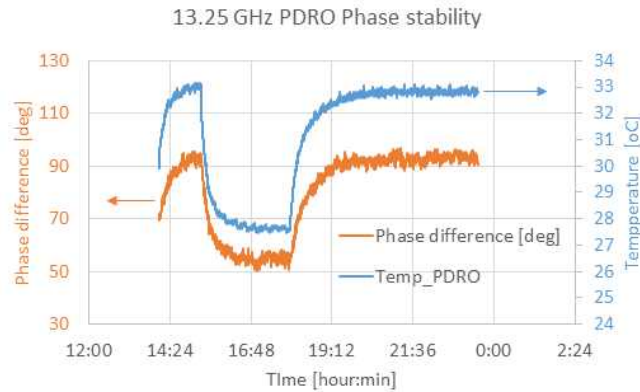
현재 세종 VLBI 망원경의 22/43GHz 수신기의 위상 안정도를 평가하기 위해서 22GHz 수신기에 사용된 것과 동일한 13.25GHz PDRO(Phase Locked Dielectric Resonator Oscillator)의 위상 잡음과 위상 안정도를 실험실에서 측정하였다. 그림 6-6과 6-7에 측정 구성도와 사진을 보였다. 그림 6-7의 팬을 이용하여 Test용 PDRO의 온도 변화를 일으켜서 온도에 따른 PDRO의 위상 변화를 측정하였다. 5 °C의 온도 변화에 40 degree의 위상 변화가 나타나서 8 deg/°C의 온도 변화에 따른 위상 안정도를 보였다.



<그림 6-6> 13.25GHz PDRO의 온도에 따른 위상 안정도 측정 setup

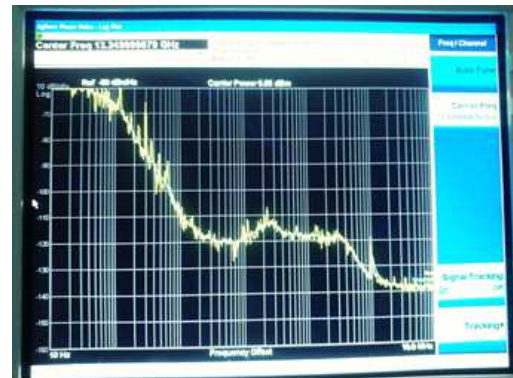
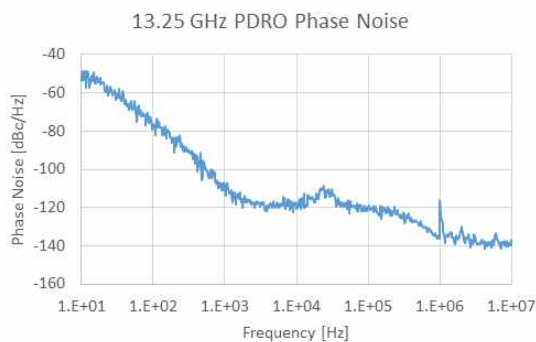


<그림 6-7> 13.25GHz PDRO의 온도에 따른 위상 안정도 측정 사진; 왼쪽은 Reference 용 13.25GHz PDRO와 위상 안정도 비교를 위한 부품 구성 사진, 오른쪽은 Test용 PDRO 와 온도 변화를 위한 팬 설치 사진



<그림 6-8> 13.25GHz PDRO의 온도에 따른 위상 안정도 측정 결과; 8 deg/°C의 온도에 따른 위상 안정도를 보임.

13.25GHz PDRO의 위상잡음은 1 초 이하의 짧은 시간동안의 위상 안정도를 나타내는 양으로 그림 6-9의 왼쪽 그림과 같이 100 Hz offset 주파수에서 -75 dBc/Hz, 100KHz offset 주파수에서 -120 dBc/Hz 이하의 매우 우수한 위상 잡음을 보였다. 망원경 수신기실에서 측정한 위상잡음은 100 Hz offset 주파수에서 -68 dBc/Hz, 100KHz offset 주파수에서 -118 dBc/Hz 로서 낮은 offset 주파수에서 더 큰 차이를 보였다. 이는 안테나 수신기실의 기준신호 발생기에서 공급하는 100MHz 신호의 위상 잡음이 다소 높기 때문이다. 하지만 안테나 수신기실의 위상잡음도 전체 수신기 안정도에 1 degree 이하로 매우 작은 영향을 줄 수 있다.

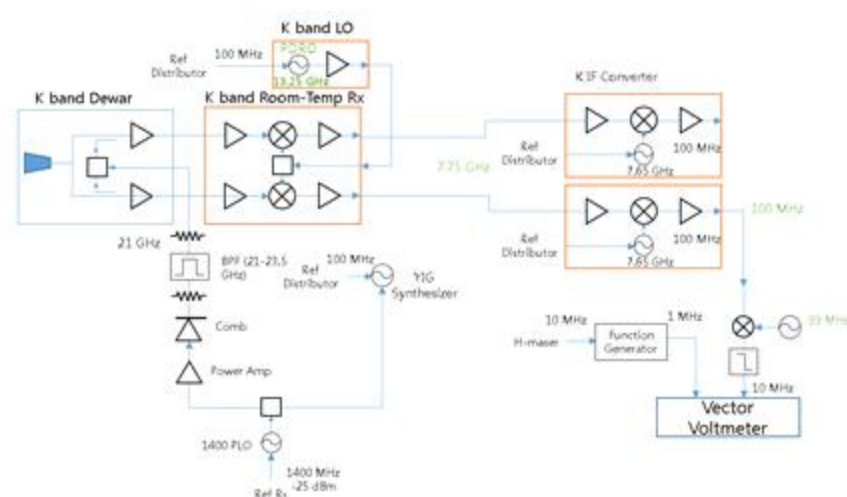


<그림 6-9> 13.25GHz PDRO의 100MHz 기준 주파수 신호에 따른 위상잡음 측정 결과; (좌) 실험실 측정 결과,(우) 안테나 수신기실 측정 결과, 200 Hz 이하의 offset 주파수에서 100MHz 기준 주파수 신호의 위상 잡음의 영향을 받음.

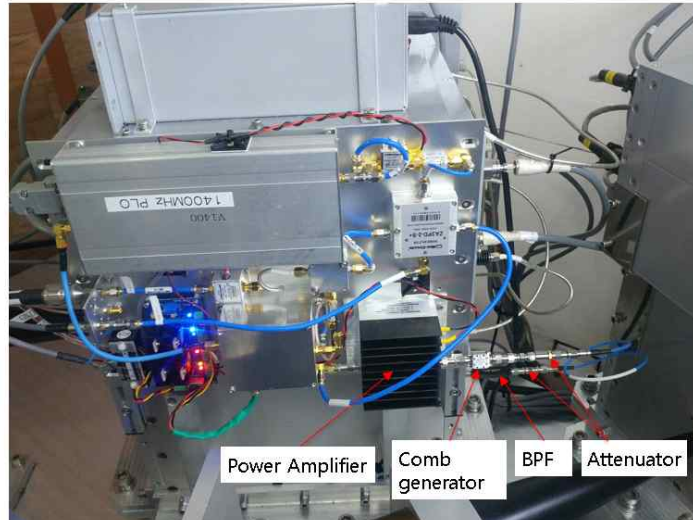
22GHz 수신기를 이용한 관측에서 얻어지는 위상 안정도에는 13.25GHz PDRO 외에 주파수 하향 변환기에서 사용하는 7.65GHz PDRO 도 영향을 미친다. 7.65GHz PDRO는 13.25GHz PDRO와 같은 100MHz 기준 주파수를 사용하고, 출력 주파수가 낮아서 절반 정도의 온도에 따른 위상 안정도를 나타낼 것으로 예측할 수 있다.

22GHz 수신기와 주파수 하향 변환기의 온도에 따른 위상 안정도 측정을 위하여 1.4GHz 신호의 하모닉 주파수로서 21GHz를 생성하여 전체 수신기의 온도에 따른 위상 안정도를 점검하였다. 그림 6-10에 수신기 위상 안정도 측정 구성도를 보였다. 1.4GHz PLO(Phase Locked Oscillator) 회로의 출력 하모닉 신호를 수신기의 입력 coupling 포트에 인가하여 관측기기실에 전송된 100MHz IF 신호를 광 컨버터를 통해 관측기기실로 보내서, H-maser 기준신호와 비교하여 그림 6-11~13에 보였다. 안테나의 고도를 0 도에서 80도, 다시 80도에서 0 도로 움직이면서 안테나 수신기실의 온도 변화를 만들었다. 안테나 고도에 따라 온도가 1.5℃ 변할 때 위상은 40 deg 변하여 25 deg/℃ 이상의 온도 영향을 보인다. 이는 PDRO 만의 온도에 따른 위상 안정도인 8 deg/℃ 보다 3배 정도 높은 값이다. 이는 IF converter의 7.65GHz 국부발진기 외에도 기준신호 발생기에서 1.4GHz 신호에서 100MHz를 발생시키는 장치의 안정도가 영향을 미친 것으로 판단된다.

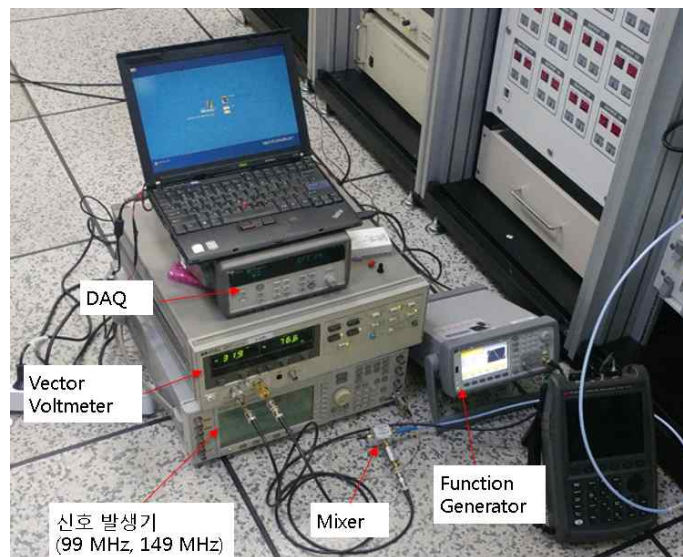
43GHz 수신기의 안정도는 수신기에 사용된 17.055GHz 국부발진기와 주파수 doubler의 사용을 고려하면, 22GHz 수신기의 안정도 보다 2 배 나쁜 50 deg/℃ 이상의 온도 안정도를 나타낼 것으로 보인다.



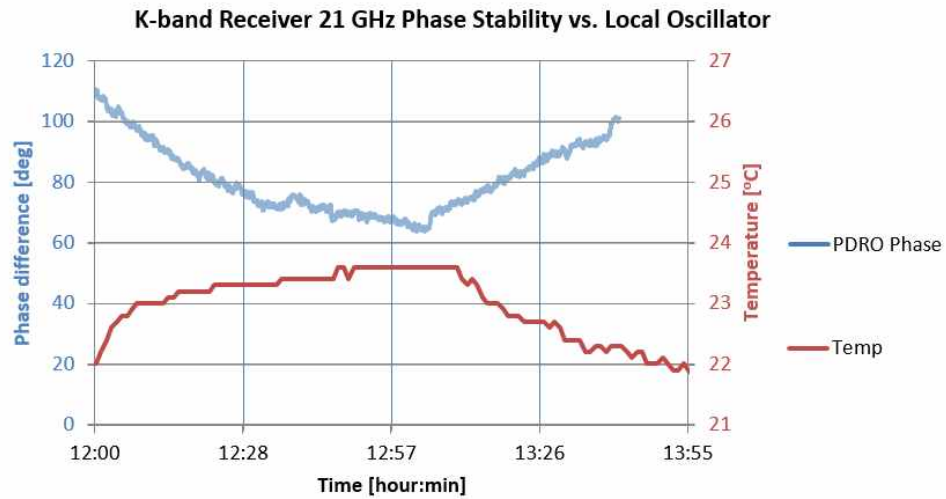
<그림 6-10> 기존의 국부발진기를 사용한 22GHz 수신기 위상 안정도 측정 구성도; 1.4GHz PLO 회로의 출력 하모닉 신호를 수신기의 입력 coupling 포트에 인가하여 100MHz IF 신호를 H-maser 기준신호와 비교하여 전체 아나로그 수신 시스템의 위상 안정도를 측정.



<그림 6-11> 22GHz 대역 수신기에 1.4GHz 하모닉 신호를 인가하기 위해 1.4GHz PLO 의 출력 신호를 Power Amplifier와 Comb generator 등에 수신기의 Coupling 포트에 21GHz 신호를 공급하였음.



<그림 6-12> IF 100MHz 신호와 H-maser 신호의 위상 비교를 위한 측정 setup



<그림 6-13> 21GHz 신호를 인가했을 때의 22GHz 대역 수신기의 온도에 따른 위상 안정도 측정 결과; 온도가 1.5°C 변할 때 위상은 40 deg 변하여 25 deg/°C 이상의 온도 영향을 보임.

2. 주파수 합성기 개발을 위한 요구 성능 설계

가. 주파수 합성기 출력 주파수 범위, 해상도, 출력 전력 계산

현재의 세종 망원경의 22/43GHz 수신기의 주파수 대역은 21-23/42-44GHz 이고, 국부 발진기 주파수로 13.25/34.11GHz를 사용한다. 수신기 광대역화를 위한 관측 주파수 대역으로는 18-26/33-50GHz를 고려할 수 있다. 광대역 수신기의 국부발진 주파수는 IF(Intermediate Frequency) 주파수 범위에 따라 결정된다. 현재와 동일한 IF 주파수 대역(8-10GHz)를 갖는 방법과 현재 도입된 DBBC3의 입력 주파수 대역(4-15GHz)에 따르는 방법이 있다.

먼저 현재의 IF 주파수 대역인 8-10GHz를 사용할 경우에는 넓은 주파수 범위의 국부 발진기가 요구된다. 즉, 18-26GHz 수신기는 10-16GHz, 33-50GHz 수신기는 25-40GHz 대역의 국부 발진 주파수를 선택할 수 있다. 25-40GHz 국부 발진기는 주파수 체배기(frequency doubler)를 포함할 경우, 12.5-20GHz 대역의 주파수 합성기가 필요로 된다. 2GHz 이상의 대역폭을 갖는 광대역 e-VLBI backend 시스템을 사용할 경우, 출력 주파수 간격은 2GHz 로도 충분할 것이다. 이상에서 설정한 주파수 대역을 아래 표에 정리하였다. 아래 표 6-1로부터 22/43GHz 수신기를 모두 만족하는 주파수 합성기의 주파수 대역과 간격은 10-20GHz, 0.5GHz 이다.

<표 6-1> 현재 수신기의 8-10GHz IF 주파수를 사용할 경우, 수신기 광대역화에 요구되는 주파수 합성기의 주파수 범위와 간격

항목	22GHz 수신기	43GHz 수신기
관측 주파수 대역	18-26GHz	33-50GHz
IF 주파수 대역	8-10GHz	8-10GHz
국부 발진기 주파수 범위	10-16GHz	25-40GHz
국부 발진기 주파수 간격	1GHz	1GHz
주파수 합성기 주파수 범위	10-16GHz	12.5-20GHz
주파수 합성기 주파수 간격	1GHz	0.5GHz

IF 주파수 대역을 DBBC3의 입력 주파수 대역인 4-15GHz를 사용할 경우에는 한, 두 개의 국부 발진기 주파수로 22, 43GHz 수신기의 전체 주파수 대역을 동시에 관측할 수 있다. 다음

의 표 6-2, 3에 DBBC3에 맞는 주파수 변환 방법을 정리하였다. 22GHz 수신기의 18-26GHz는 단일 국부발진기 주파수로 주파수 변환이 가능하다. 43GHz 수신기의 33-50GHz 주파수 대역은 Power divider를 거쳐서 2개로 나눈 후에 33-42GHz는 29GHz 국부발진기 주파수로 RF 신호가 USB(Upper Side Band)에 오도록 주파수 변환을 하고, 41-50GHz는 54GHz 국부발진기 주파수로 RF 주파수가 LSB(Lower Side Band)에 오도록 주파수 변환을 한다.

<표 6-2> DBBC3 입력 주파수 대역인 4-15GHz IF 주파수를 사용할 경우, 수신기 광대역화에 요구되는 주파수 합성기의 주파수 범위와 간격

항목	22GHz 수신기	43GHz 수신기
관측 주파수 대역	18-26GHz	33-50GHz
IF 주파수 대역	4-12GHz	4-13GHz
국부 발진기 주파수	14GHz, 고정	28, 54GHz
주파수 합성기 주파수	14GHz	14, 18GHz

<표 6-3> 43GHz 대역 수신기의 주파수 할당 방법

RF [GHz]	IF [GHz]	LO [GHz]	Syn [GHz]	remark
33	5	28	14	USB
34	6	28	14	USB
35	7	28	14	USB
36	8	28	14	USB
37	9	28	14	USB
38	10	28	14	USB
39	11	28	14	USB
40	12	28	14	USB
41	13	28	14	USB
41	13	54	18	LSB
42	12	54	18	LSB
43	11	54	18	LSB
44	10	54	18	LSB
45	9	54	18	LSB
46	8	54	18	LSB
47	7	54	18	LSB
48	6	54	18	LSB
49	5	54	18	LSB
50	4	54	18	LSB

현재 세종 VLBI 22/43GHz 수신기의 정확한 1st IF 주파수 대역은 2.5GHz이고, DBBC3의 DC-4GHz 대역의 입력 주파수 대역을 사용하고 있지만, BBC(BaseBand Converter)의 좁은 IF 주파수 대역(0.5-1GHz)로 인해 2.5GHz 전체 대역을 동시에 관측하지 못하고 있다. 수신기 주파수 대역을 18-26/33-50GHz로 광대역화를 하지 않더라도 현재 수신기의 대역폭을 최대한 활용하기 위해서는 BBC의 IF 주파수 대역을 확장할 필요가 있다. 또한 RF 대역이나 IF 대역에 있는 BPF(Band Pass Filter)를 교체하거나 수신기의 국부발진 주파수를 2GHz 정도 범위에서 조정할 수 있으면, 현재 수신기의 구성을 최소한으로 바꾸면서 가능한 넓은 범위의 관측이 가능할 것이다.

주파수 합성기 출력 전력 규격은 주파수 체배기나 주파수 혼합기 규격의 영향을 받으며, 대략 10-15 dBm 범위에서 정해질 것이다.

나. 22/43GHz 수신기의 관측 효율을 고려한 주파수 별 필요 위상 안정도 계산

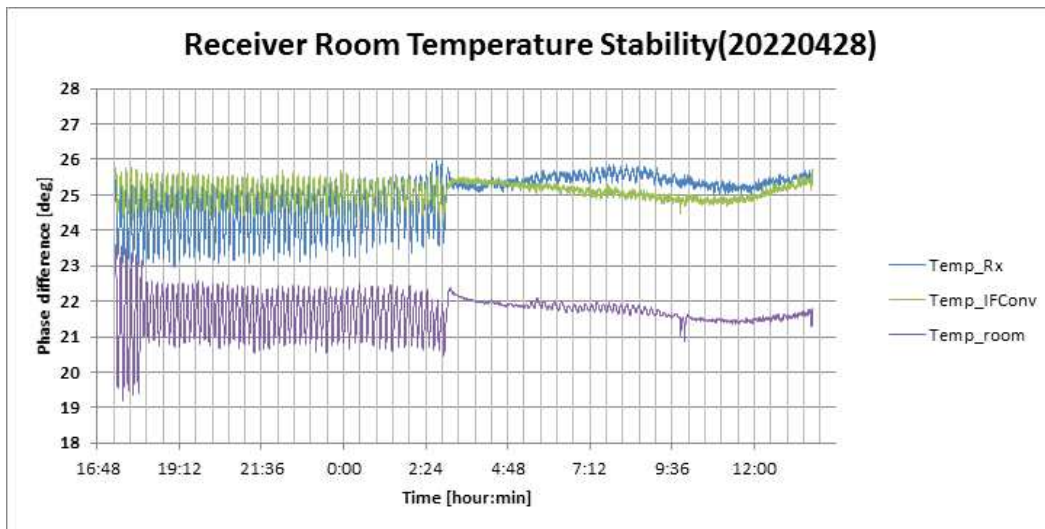
주파수 합성기의 위상 잡음과 온도에 따른 위상 변화 규격 결정은 22/43GHz 대역 관측 효율, 공급되는 기준 주파수 신호의 위상 잡음, 안테나 수신기 실의 온도 안정도의 영향을 고려해야 한다. 국부발진기의 위상 잡음으로 인해 5% 상관 효율 감소가 되려면, 43GHz에서 18 deg 이하의 위상잡음이 요구된다[8]. 22/43GHz 동시관측을 위해서는 22GHz에서 9 deg 이하의 위상잡음이 필요하다. 1 deg rms 오차를 주파수합성기의 주파수 offset에 따른 위상잡음은 아래 표 6-4와 같다. 이 값은 현재 망원경에서 설치된 PDRO와 유사한 위상잡음으로서 주파수합성기에서도 충분히 얻을 수 있다.

<표 6-4> 10GHz 대역 주파수 합성기의 위상잡음(1 deg rms의 경우)

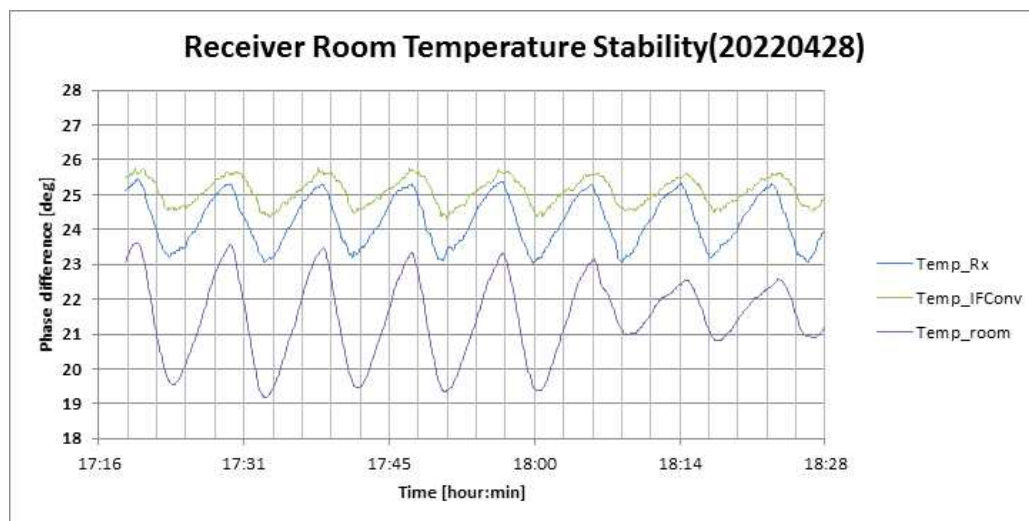
Frequency Offset	위상 잡음 [dBc/Hz]
100 Hz	-70
1 kHz	-85
10 kHz	-95
100 kHz	-100
1 MHz	-115

온도에 따른 국부발진기의 위상 안정도가 관측 효율에 주로 영향을 미칠 것으로 보인다. 안테나 수신기 실에서 1℃의 온도 변화를 가정할 경우, 앞에서 얻어진 22GHz 수신기의 위상 안정도인 25 deg/℃는 5% 관측 효율에 해당되는 9 deg/℃를 2배 이상 벗어나는 값으로 개선이 필요하다. 13.25GHz PDRO와 7.65GHz PDRO의 안정도 개선도 필요하지만, 기준신호발생기에서 1.4GHz 신호에서 100MHz 발생기의 영향을 줄여야 한다. 이를 위해서는 기준신호발생기의 1.4GHz 기준 신호를 직접 사용하여 13.25GHz와 7.65GHz를 만들어야 한다. ‘

다음의 그림 6-14, 15에 2022년 4월 28-29일에 측정한 안테나 수신기실의 온도 변화를 측정한 결과를 보였다. 안테나의 고도가 5 deg 상태에서 이를 동안 측정하였다. 안테나 수신기 실의 온도는 외부 온도의 영향을 받아 최대 4도의 온도 변화를 보였다. 수신기 근처에서는 3분의 짧은 시간동안 최대 2도의 온도 변화를 보였다. 외부 온도가 높을 경우, 날씨에 따라 항온습기의 compressor가 동작하여 9분 정도의 주기로 온도가 최대 2-4도 변하지만, 한 주기 동안의 평균 온도의 변화는 수 시간 동안 1도 이내로 유지되었다. 안테나의 고도에 따른 온도 변화는 그림 13에서 볼 수 있듯이, 1-2℃로 측정되었다. 항온습기가, compressor가 꺼진 상태로 동작할 때는 수 시간동안 1도 이내로 안정된 상태를 보였다. 실제 국부발진기가 설치된 금속 고정면의 온도는 실내 온도처럼 빠르게 변하지는 않을 것이다. 이상의 결과로부터 국부발진기가 설치되는 금속면이나 상자의 온도가 방 온도보다 천천히 변화하도록 할 필요가 있다.



<그림 6-14> 2022년 4월 28-29일에 이루어진 안테나 수신기 실의 온도 변화; 안테나 고도가 0도 일 때 수신기 근처(Temp_Rx), BBC 근처(Temp_IFConv), 수신기 실 한가운데(Temp_room)에서의 오후 5시 19분부터 다음 날 오후 1시 43분까지의 온도 변화



<그림 6-15> 짧은 시간 동안의 항온항습기 동작에 따른 수신기 실 내부의 온도 변화; 안테나 고도가 0도 일 때 수신기 근처(Temp_Rx), BBC 근처(Temp_IFConv), 수신기 실 한가운데(Temp_room)에서의 오후 5시 19분부터 6시 28분까지의 온도 변화

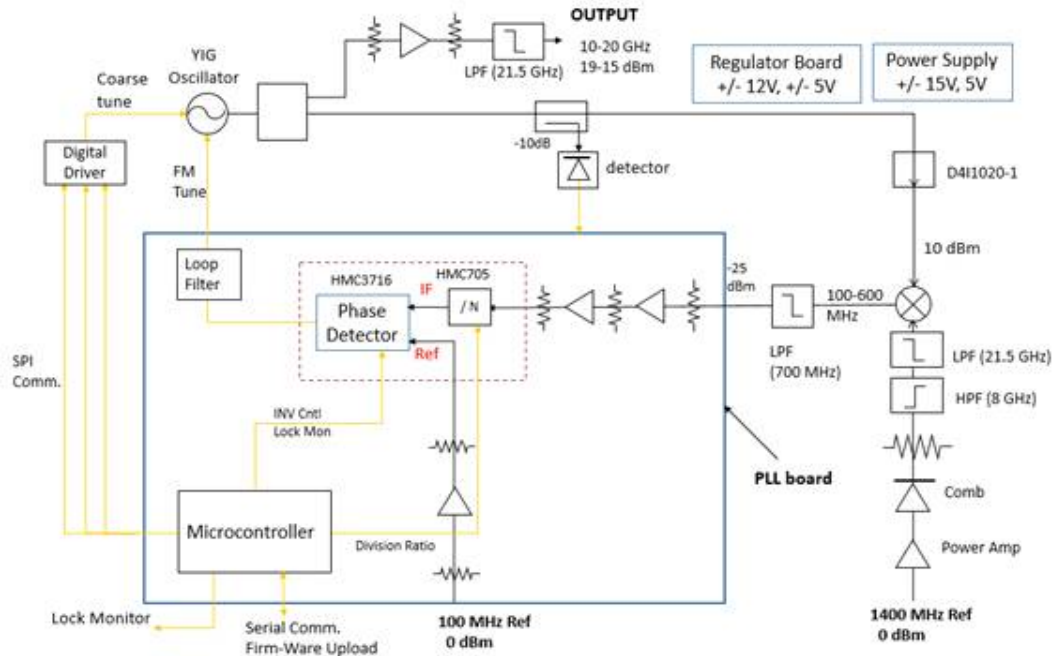
3. 주파수 합성기 회로 설계 및 제작

가. 주파수 합성기 시스템 설계

주파수 합성기는 1.4GHz 이상의 높은 입력 기준 주파수 신호의 하모닉 주파수와 주파수 발생기(Oscillator) 신호를 혼합하여 600MHz 이하의 낮은 IF 주파수로 변환하여, 100MHz 기준 주파수 신호와 위상 비교를 하는 방식으로 그림 6-16과 같이 설계되었다. 18-26/33-50GHz 관측을 위해 8-10GHz IF 주파수를 사용할 경우에 대비하여, 10-20GHz 대역 광대역 주파수 혼합기를 설계하였다. 출력 주파수는 아래 식에서 M이 7-14, N이 1-6 사이의 정수일 때, 아래 식의 관계로부터 정해진다. 출력 주파수 간격은 주로 100MHz이며 최대 200MHz이다. 전체 출력 주파수 목록은 표 6-5에 보였다.

$$F_{OUT} = (1.4 \times M \pm 0.1 \times N) \text{ GHz}$$

출력 주파수와 기준 주파수 사이의 차이가 10배 정도로 작아서 10MHz나 100MHz 기준 주파수를 사용하는 상용 주파수 발생기와 비교하여, 출력 신호의 위상이 온도 변화의 영향을 덜 받는다. 위상 잡음도 1.4GHz 기준 주파수와 $20 \log_{10}(M)$ 으로 주어져서 1.4GHz 신호의 위상 잡음의 영향을 받게 된다.



<그림 6-16> 1.4GHz와 100MHz 기준 주파수 신호를 사용한 10-20GHz YIG 주파수 합성기 구성도

<표 6-5> 10-20GHz YIG 주파수 합성기의 출력 주파수 목록

1.4GHz harmonic number(M)	IF Frequency($\pm 0.1 \times N$) [GHz]											
	-0.6	-0.5	-0.4	-0.3	-0.2	-0.1	0.1	0.2	0.3	0.4	0.5	0.6
	Synthesizer Output Frequency [GHz]											
7								10	10.1	10.2	10.3	10.4
8	10.6	10.7	10.8	10.9	11	11.1	11.3	11.4	11.5	11.6	11.7	11.8
9	12	12.1	12.2	12.3	12.4	12.5	12.7	12.8	12.9	13	13.1	13.2
10	13.4	13.5	13.6	13.7	13.8	13.9	14.1	14.2	14.3	14.4	14.5	14.6
11	14.8	14.9	15	15.1	15.2	15.3	15.5	15.6	15.7	15.8	15.9	16
12	16.2	16.3	16.4	16.5	16.6	16.7	16.9	17	17.1	17.2	17.3	17.4
13	17.6	17.7	17.8	17.9	18	18.1	18.3	18.4	18.5	18.6	18.7	18.8
14	19	19.1	19.2	19.3	19.4	19.5	19.7	19.8	19.9	20		

주파수 발생기로는 매우 낮은 위상잡음을 갖는 YIG 주파수 발생기를 사용하였다. YIG 주파수 발생기는 Coarse tune과 FM tune 기능을 별도로 갖고 있다. Coarse tune을 사용하여 원하는 출력 주파수 근처에 YIG 출력 주파수가 나오도록 하고, FM tune 제어와 PLL 회로, 100MHz 기준 주파수를 사용하여, 주파수 동기가 되도록 하였다. 제조사에서는 Coarse tune 제어를 직렬 통신 명령으로 가능하게 하는 직렬 드라이버를 제공한다. 제조사의 직렬 통신 드라이버를 포함한 YIG 주파수 발생기의 테스트 결과와 외형을 그림 6-17과 6-18에 보였다.

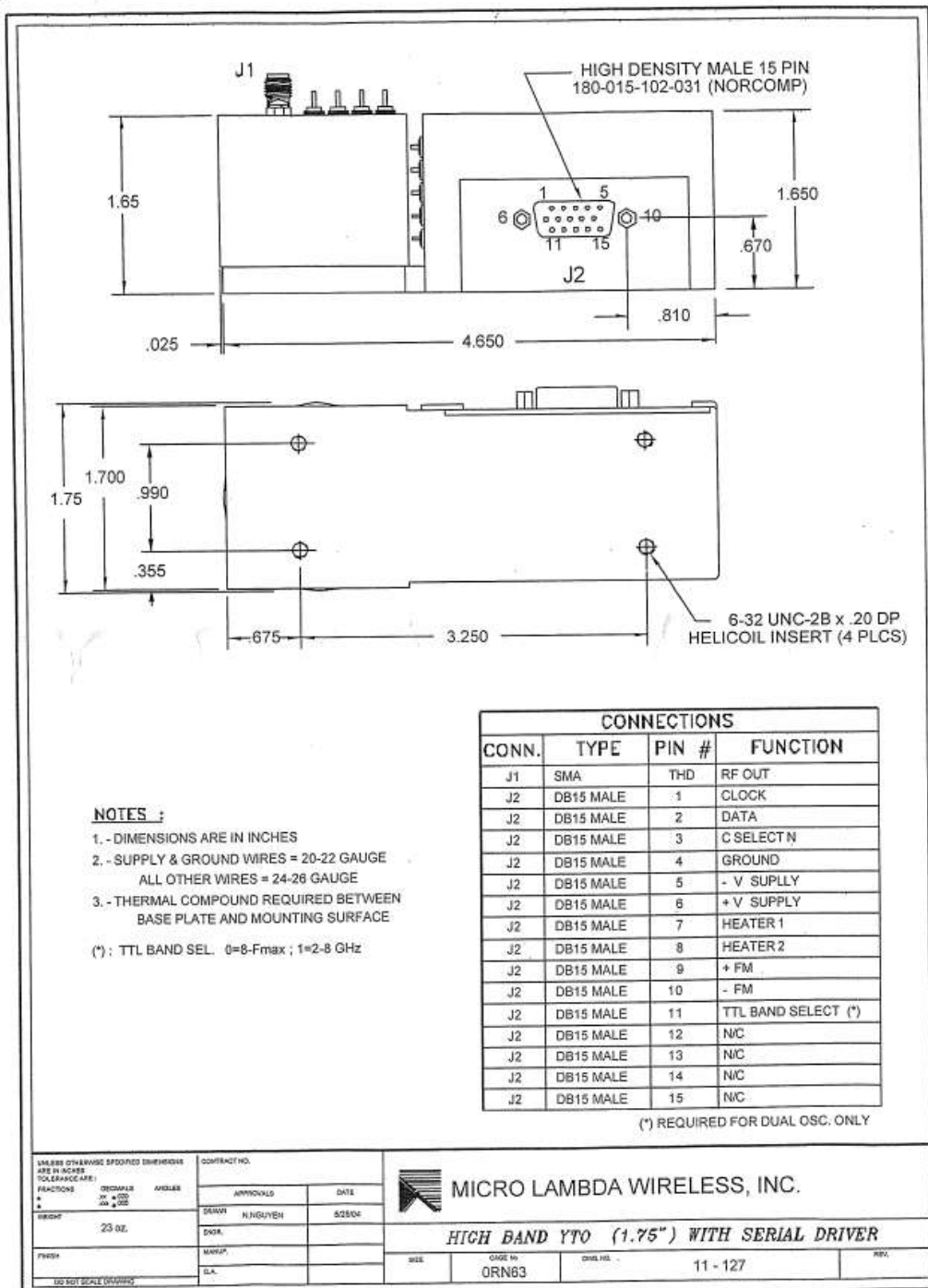
OSCILLATOR FINAL TEST DATA

MODEL No: MLXS-1838SD CUSTOMER: GMB Partner OPER: DS DATE: 12/13/21
 SERIAL No.: 28405 CUSTOMER PN: _____ REV. _____ QA: 11 DATE: 12/13/21

FREQUENCY RANGE 10.0-20.0 GHz FM SENSITIVITY 409 kHz/mA
 POWER OUTPUT +13 dBm Min FM DEVIATION ± 40.9 MHz @ 0.1A
 PW VAR. vs FREQ 6 dB Max
 HYSTERISIS 15.0 MHz SUPPLY CURRENT:
 LINEARITY $\pm 0.15\%$ MHz @ +15V 1200 mA
 HARMONICS -12 dBc @ -15V 33 mA
 THIRD HARMONIC -20 dBc HEATER: +24 V
 SPURIOUS -70 dBc SURGE: 300 mA
 STEADY: 24 mA
 PHASE NOISE:
 10-18 GHz @ 100 kHz: -125 dBc/Hz min.
 18-20 GHz @ 100 kHz: -123 dBc/Hz min.

CONTROL (HEX)	FREQ. (MHz)	ΔF (MHz)	POWER (dBm)
0000	10000	0	19.2
8000	15000	3	15.1
FFFF	20000	0	15.6

<그림 6-17> 10-20GHz YIG 주파수 발생기(MLXS-1838SD)의 테스트 결과

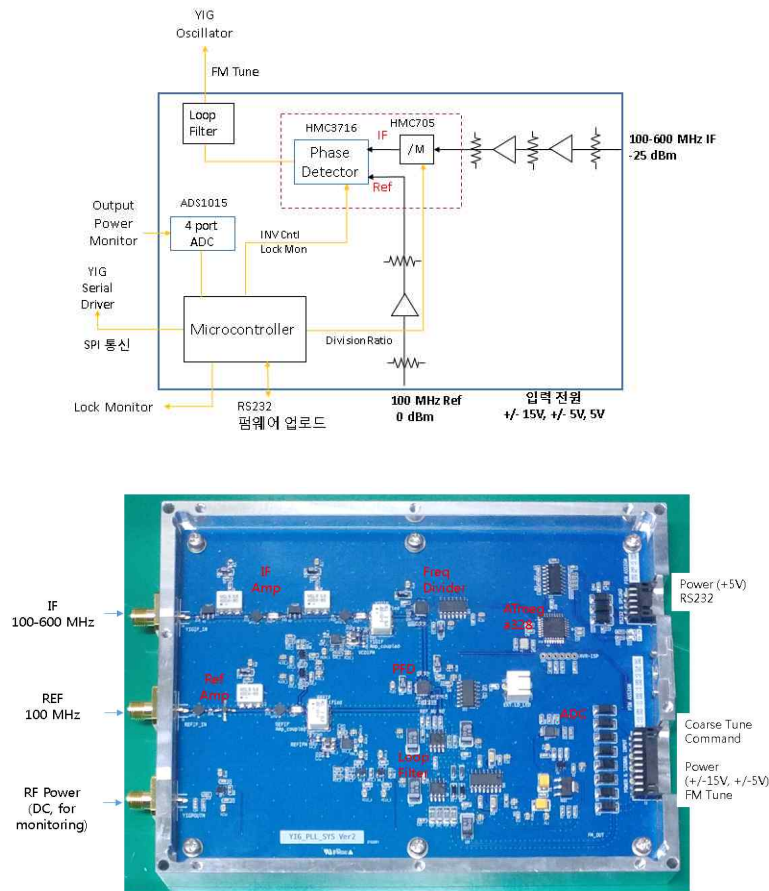


<그림 6-18> 10-20GHz YIG 주파수 합성기의 직렬통신 포트 핀 할당과 외형

나. PLL회로 설계 및 제작

주파수 합성기 구성도에서 PLL 회로 부분을 그림 19(상)에 보였다. PLL 회로는 12-18GHz 대역폭을 갖는 KVN 광대역 주파수합성기와 동일한 회로를 사용하였다. 1.4GHz 하모닉 신호와 YIG 발진기 신호의 차 주파수인 100-600MHz IF 대역 신호는 2개의 증폭기

를 거쳐서 Analog devices 사의 HMC705 주파수 분주기에 입력되어 IF 주파수에 따라 1-6의 주파수 분주비를 적용하여 100MHz를 만든다. HMC705 출력 100MHz 신호는 100MHz 기준 신호와 HMC3716 위상 주파수 검출기에서 위상 차에 비례하는 전압을 만들어 Loop Filter를 거쳐서 YIG 주파수 발생기의 FM tune 포트에 공급한다. Loop Filter는 주파수 분주비에 따른 Loop Bandwidth의 차이를 줄이기 위해서, 2개의 op amp 회로 중에서 선택하도록 하였다. 주파수 분주비가 1 일 때는 Loop Bandwidth가 800KHz, 분주비가 6 일 때는 300KHz의 Loop bandwidth를 갖는다. 그림 6-19(하)에 제작된 PLL 보드 사진을 보였다.

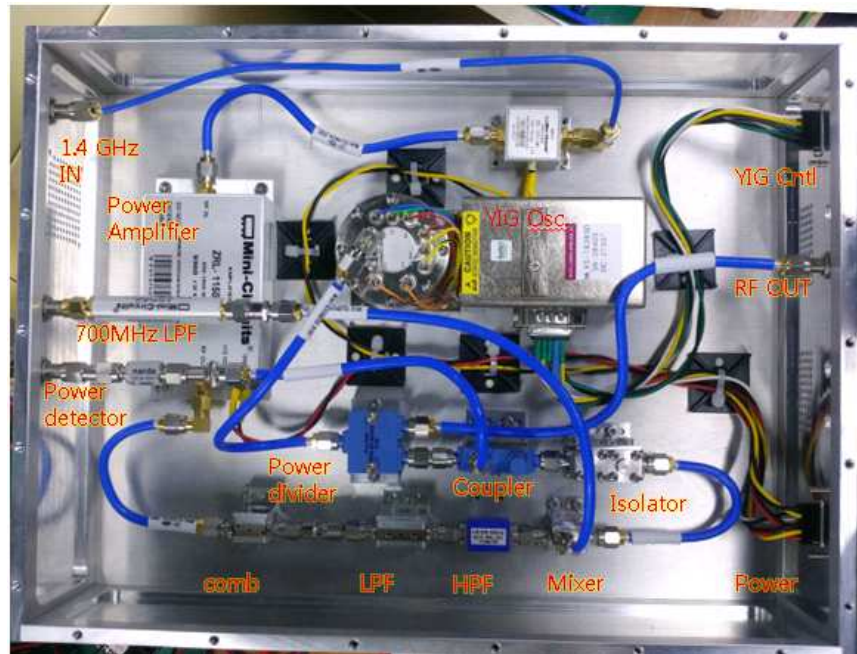


<그림 6-19> (상) PLL 회로 구성도. (하) 제작된 PLL 회로

다. YIG 발진기를 활용한 RF 회로 설계 및 제작

RF 회로는 YIG 발진기를 포함한 신호 발생 부분과 위상 비교를 위한 1.4GHz 하모닉 신호 발생을 위한 부분, 주파수 혼합기와 LPF 부분으로 구분된다. 그림 6-16에서 PLL 회로를 제외한 나머지 부분이 RF 회로에 해당된다. 그림 16에서 YIG 발진기 출력 신호는 power divider에서 2개로 분리되어 하나는 수신기 국부발진기로 사용되고, 다른 하나는 주파수 동

기를 위해서 주파수 혼합기에 공급된다. 1.4GHz 기준 신호는 전력 증폭기에서 증폭되어 하모닉 신호 발생을 위해 comb generator에 전달된 후 Mixer의 RF 포트에 연결된다. Mixer에서는 공급된다. 그림 20에 알루미늄 상자에 조립된 RF 회로를 보였고, 표 6-6에 RF 부품 목록을 정리하였다.



<그림 6-20> RF 부품들이 조립된 알루미늄 상자 내부

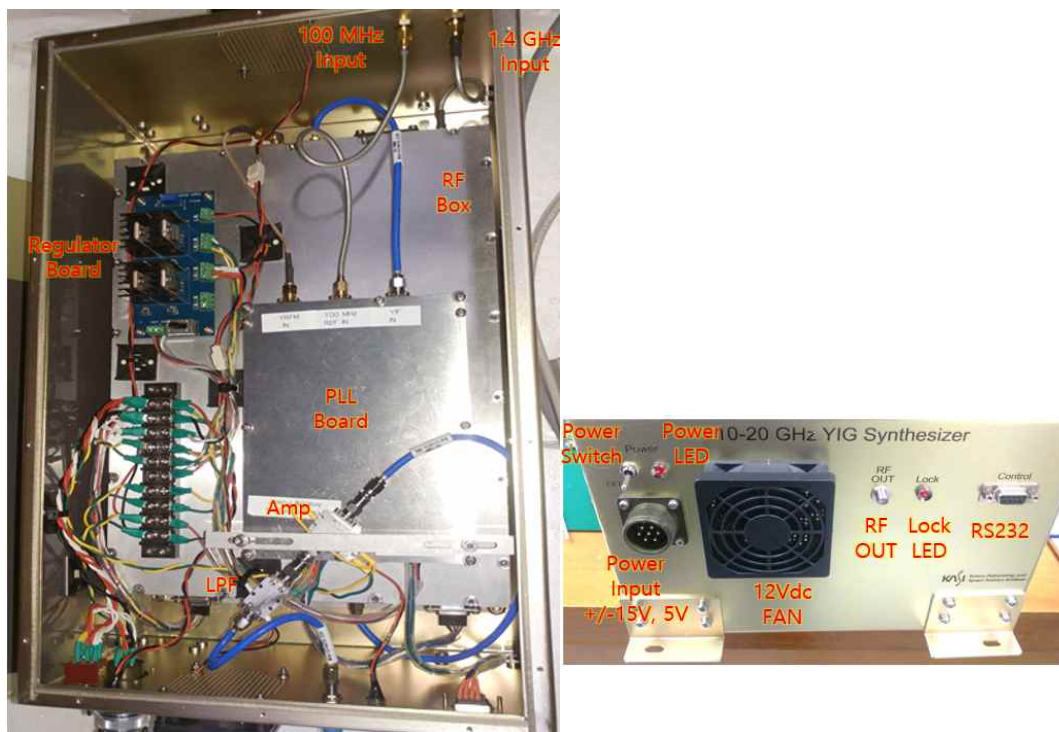
<표 6-6> RF 회로의 주요 부품 목록

부품 명	모델명	제조사	규격
YIG 발진기	MLXS-1838SD	Microlambda Wirelss	10-20GHz, 10 dBm
Power Amplifier	ZRL-1150LN+	Minicircuits	650-1400MHz, P1dB=25 dBm(typ.)
Comb generator	NLTL6273	Marki Microwave	-20 ~ -25 dBm @ 10-20GHz
LPF	FLP-2150	Marki Microwave	3dB cutoff: 21.5GHz
HPF	ZHSS-8G-S+	Minicircuits	1dB cutoff: 8GHz
Mixer	MM1-0832LS	Marki Microwave	RF/LO : 8-32GHz IF : DC-12GHz
Power Divider	PS2-50-435/8S	Pulsar Microwave	4-27GHz
Coupler	CS10-50-435/4	Pulsar Microwave	6-26.5GHz
Isolator	D4I1020	DITOM	10-20GHz, 32 dB isolation
700MHz LPF	VLFX-650	Minicircuits	1dB cutoff: 650MHz
Power Detector	4506	Narda Microwave	10MHz - 18GHz

라. 주파수 차폐 상자를 포함한 전체 주파수 합성기 조립

RF 부품이 포함된 알루미늄 상자 내부에는 1.4GHz comb generator가 있어서 여러 주파수 대역의 신호들이 외부로 누설될 수 있다. 이렇게 누설된 신호는 22/43GHz 대역 수신기나 2/8GHz 수신기에 간섭 신호로 나타날 수 있어서 RF 상자를 직육면체로 제작하여 외부 전파 누설을 막았다. 추가로 RF 상자와 PLL 보드 등을 다시 알루미늄 차폐 상자 내부에 들어가도록 하여, 이중으로 간섭으로 인한 전자파 누설을 차단하였다.

전체 주파수 합성기의 전면에는 Power(+/-15V, +5V) 입력과 RS232 통신 포트, RF 출력 포트, Power 및 Lock LED가 설치되어 있다. 전면과 후면에는 전동 팬을 설치하여 공기가 외부와 통하도록 하였다. 후면에는 100MHz와 1.4GHz 기준 주파수 신호가 입력이 된다.



<그림 6-21> 주파수 합성기 차폐상자 내부와 전면 사진

마. 주파수 합성기 프로그램 작성 및 주파수 합성기 완성

주파수 합성기 프로그램은 표 6-7의 기능을 수행하도록 작성되었다. PLL 보드에 사용된 Micro-controller는 Arduino Uno용 Atmega a328이다. 아두이노용 소프트웨어를 사용하여 프로그램을 작성하였다. 설정 주파수 명령은 RS232 포트를 통해 주파수를 100MHz 단위로 3개의 숫자를 보내면 된다. 주파수 합성기에서는 설정 주파수를 16 비트 코드로 바꾸어서 YIG 발진기용 Serial driver에 전달한다. 작성된 프로그램은 부록에 수록하였다.

<표 6-7> 주파수 합성기 프로그램의 주요 기능 및 기능 설명

번호	기능	기능 설명
1	주파수 명령 받기	100MHz 단위로 3 자리의 주파수 명령을 RS232 통신을 통해서 받음
2	YIG Coasre tune	주파수에 해당되는 16비트 코드를 YIG Driver에 전달, 10000MHz: 0x0000, 20000MHz:0xFFFF
3	HMC705 주파수 분주비 설정	100-600MHz IF 주파수를 100MHz로 분주
4	HMC3716 INV 설정	YIG 출력 주파수가 1.4GHz 하모닉 주파수보다 크면 0,작으면 1 설정
5	HMC3716 Lock 모니터	Lock이 되어 있으면 1, Unlock 이면 0 이 모니터 됨.
6	ADC 값 읽기	YIG FM tune 전압,IF 신호 크기, 100MHz 기준 신호 크기, RF 신호 크기

주파수 명령을 받으면, 주파수 합성기에서는 설정 주파수, 주파수 코드, FM tune 전압, IF 신호 크기, 100MHz 기준 신호 크기, RF 신호 크기, Lock detect 값을 순서대로 보내준다. 다음의 표 8에 모니터 되는 값의 의미와 정상 범위에 대해 기술하였다. 가장 중요한 모니터 값은 FM tuning 전압이다. 정상 범위에서 벗어나면 주파수 Lock이 되지 않는다. IF 신호와 100MHz 기준 신호의 크기 모니터 응답 값에 이상이 발견되었다. PLL 회로는 정상 작동하여, 전압을 모니터하는 ADC 연결 부분의 원인으로 추정하고 있다.

<표 6-8> 주파수 합성기 응답 명령

명령 예	명령 설명	정상 범위
13699.85	주파수 근사값, 실수	
24247	16 진수 주파수 코드 (0:10GHz, 65535:20GHz)	0-65535
2.55	FM tuning 전압	1.5-3.5
3.69	IF 신호 크기	-5dBm: 1.85, -15dBm: 1.45
0.25	100MHz 기준 신호 크기	-5dBm: 1.85, -15dBm: 1.45
0.87	YIG 출력 신호의 10 dB coupled RF 신호 크기	0.2-1
1	Lock 여부	1: 정상, 0: unlock

4. 주파수 합성기 종합성능 테스트

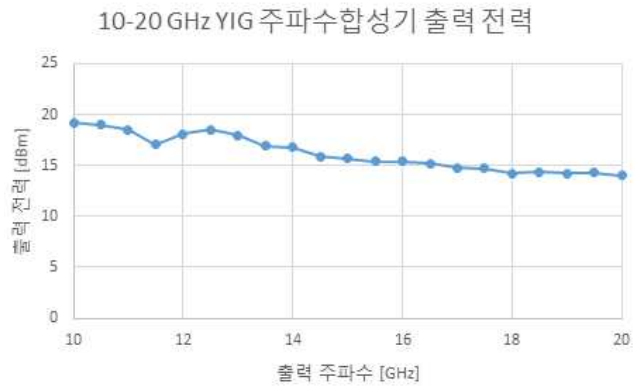
주파수 합성기가 원하는 특성을 만족하는지 보기 위해, 출력 전력, 위상 잡음, 온도에 따른 위상 안정도를 측정하였다. 표 6-9에 주파수 합성기가 만족해야 할 주요 특성과 측정 결과를 정리하였다.

<표 6-9> 주파수 합성기가 만족할 기본 특성

항목	주파수 합성기 특성	측정 결과
출력 주파수 범위와 간격	10-20GHz, 1GHz	10-20GHz, 100/200MHz
출력 전력	> 10 dBm	14-19 dBm
위상 잡음 (17GHz)	<-70 dBc/Hz @100 Hz offset <-100 dBc/Hz @100KHz offset	<-70 dBc/Hz @100 Hz offset <-100 dBc/Hz @100KHz offset
위상 안정도	10 deg/°C	< 2 deg/°C

가. 출력 주파수 신호 크기 및 Lock 여부 측정

그림 22과 같이 조립된 상태로 Power Meter/Sensor를 사용하여 출력 전력을 측정하였다. 출력 전력은 10-20GHz 대역에서 9.6-14.5 dBm으로 측정되었다. 16-18GHz의 높은 주파수 대역에서는 10 dBm 이하로 나타났다. 현재 22GHz 수신기 국부발진기로 사용 중인 13.25GHz 신호의 전력이 13 dBm이어서, 충분한 출력 전력을 얻기 위해서 전력 증폭기를 RF 상자 외부에 설치하였다. 그림 22에 RF 상자 위에 감쇄기 2개, 21.5GHz LPF와 함께 설치한 전력 증폭기를 보였다. 출력 전력도 같은 그림에 보였다. 10-20GHz 대역에서 14-19 dBm의 출력 전력이 측정되었다. 출력 주파수에서 모두 주파수 Lock이 확인되었다. 그림 23에는 전력 증폭기로 사용된 Marki Microwave 사의 APM-6849 증폭기의 주요 규격을 캡처하여 보였다.



<그림 6-22> 전력 증폭기가 추가된 주파수 합성기의 차폐 상자 내부 사진(좌)과 주파수 합성기 출력 전력(우)

Parameter	Test Conditions	Frequency	Min	Typical	Units
P _{sat} ³	5V/5V Bias, Input Driver (See footnote)	2 GHz – 20 GHz	+19	+21	dBm
		20 GHz – 29 GHz		+19	
Small Signal Gain	5V/5V bias, -25 dBm Input Power	2 GHz – 20 GHz	9	11	dB
		20 GHz – 29 GHz		10	
Input Return Loss		2 GHz – 20 GHz		15	
		20 GHz – 29 GHz		8	
Output Return Loss		2 GHz – 20 GHz		15	
		20 GHz – 29 GHz		11	
Noise Figure		2 GHz – 26.5 GHz		5	
Reverse Isolation		2 GHz – 29 GHz		41	
Collector Current ² , I _c	5V/4V	-		13	mA
	5V/5V	-		21	
	5V/6V	-		32	
Current Mirror Current, I _b	5V/4V	-		1.5	
	5V/5V	-		2.0	
	5V/6V	-		2.6	
Input IP3 (IIP3)	5V/5V bias, -15 dBm Input Power	2 GHz – 29 GHz		+10	dBm
Output IP3 (OIP3)		2 GHz – 29 GHz		+21	
Output P1dB	5V/5V bias	2 GHz – 20 GHz		+20	
		20 GHz – 29 GHz		+15	
Input Power for Saturation	5V/5V bias	2 GHz – 29 GHz		+10	dBm
Phase Noise @ 10 kHz Offset	5V/5V bias, +9 dBm Input power	2 – 29 GHz		-170	dBc/Hz

<그림 6-23> Marki Microwave 사의 APM-6849 증폭기의 주요 규격

나. 출력 주파수별 위상 잡음 측정

22GHz 수신기와 43GHz 수신기의 국부발진기 주파수 근처에서의 위상 잡음을 측정하여 그림 24에 보였다. 위상 잡음은 1 초 이하의 짧은 시간 동안의 위상(주파수) 안정도를 나타내는 양이다. 측정된 주파수 합성기의 위상 잡음은 Loop bandwidth (300–800KHz)이내에서 1.4GHz 기준 신호의 위상 잡음과 $20 \log N$ 관계를 갖었다. 즉, 좋은 위상 잡음을 얻기 위해서는 좋은 위상 잡음을 갖는 1.4GHz 기준 신호를 사용해야 한다. 그림 6-24의 결과는 실험실에서 Agilent 사의 ESG-DP 신호 발생기를 1.4GHz 발생기로 사용했을 때의 결과이다. 1 deg의 위상 잡음을 나타내는 양과 함께 10–20GHz 주파수 합성기의 측정 결과를 표 6-10에 보였다.



<그림 6-24> 13.2GHz와 17GHz에서의 주파수 합성기 위상 잡음 측정 결과

<표 6-10> 주파수 오프셋에 따른 1 deg rms의 위상 잡음과 13.2GHz와 17GHz에서 측정한 주파수 합성기의 위상 잡음 비교; 13.2GHz와 17GHz에서 모두 1 deg rms 이하의 위상 잡음을 보임.

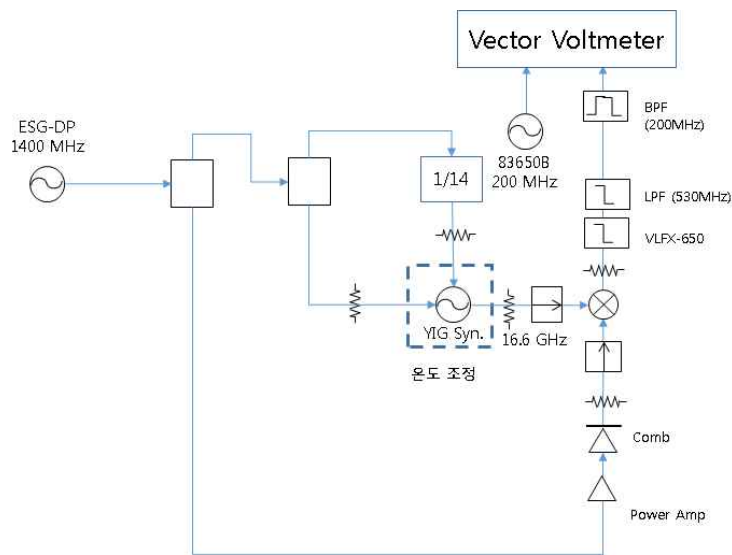
Frequency Offset	1 deg 위상 잡음 [dBc/Hz]	주파수 합성기 13.25GHz 위상잡음 [dBc/Hz]	주파수 합성기 17GHz 위상 잡음 [dBc/Hz]
100 Hz	-70	-78	-75
1 kHz	-85	-90	-88
10 kHz	-95	-106	-106
100 kHz	-100	-110	-109
1 MHz	-115	-126	-122

다. 주요 주파수 장기 위상 안정도 측정

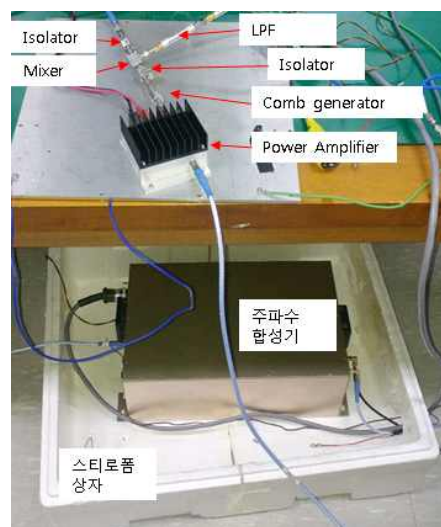
주파수 합성기의 온도에 따른 장기 위상 안정도를 측정하기 위해, 그림 25과 같은 측정 setup을 구성하였다. 기준 주파수로 사용되는 1.4GHz의 하모닉 주파수를 만들어서 위상 안정도 측정을 위한 기준 주파수로 사용하도록 하였다. 1.4GHz 하모닉 주파수와 기준 주파수 사이의 차이를 맞추기 위해서 별도로 200MHz 발생을 위한 신호 발생기(83650B)를 사용하

였다. 100MHz 기준 주파수는 1/14 분주기를 사용하여 발생시켰다.

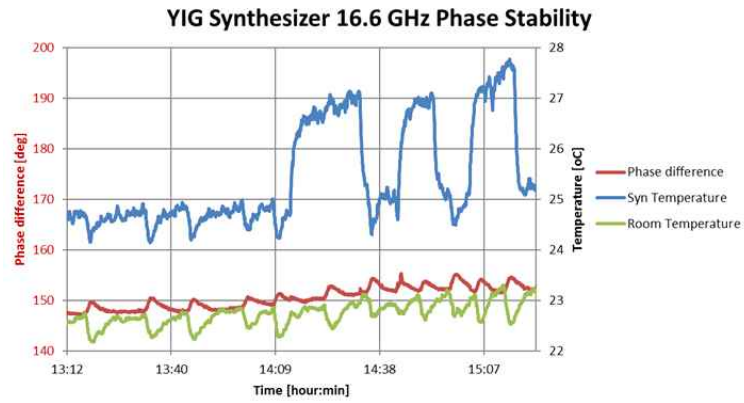
주파수 합성기의 온도 변화를 만들기 위해 그림 26과 같이 주파수 합성기를 스티로폼 상자에 넣고 뚜껑을 닫으면서 2-3 °C의 온도 변화를 발생시켰다. 온도에 따른 위상 변화는 그림 6-27에 보였다. 실험실의 온도(Room Temperature)가 0.5-1 °C 정도 주기적으로 변하면서 2-5 deg의 위상 변화(Phase difference)가 발생하였다. 스티로폼 내부의 온도 변화(Syn Temperature)에 의한 위상 변화는 보이지 않았다. 이로부터 주파수 합성기의 온도에 따른 위상 안정도는 2 deg/°C 이하로 추측된다. 더 정밀한 측정을 위해서는 실험실 온도의 유지와 1.4GHz 하모닉 발생 장치, 200MHz 신호 발생기, 1/14 주파수 분주기 등의 온도 안정도를 더 개선해야 한다.



<그림 6-25> 온도 변화에 따른 위상 안정도 측정 구성도



<그림 6-26> 온도 변화를 위해 스티로폼 상자 안에 설치한 주파수 합성기와 위상 비교를 위해 구성된 RF 부품들



<그림 6-27> 16.6GHz에서 측정된 주파수 합성기의 온도에 따른 위상 안정도; 스티로폼 상자의 온도 변화(Syn Temperature)에 따른 위상 변화는 2 deg/°C 이하임.

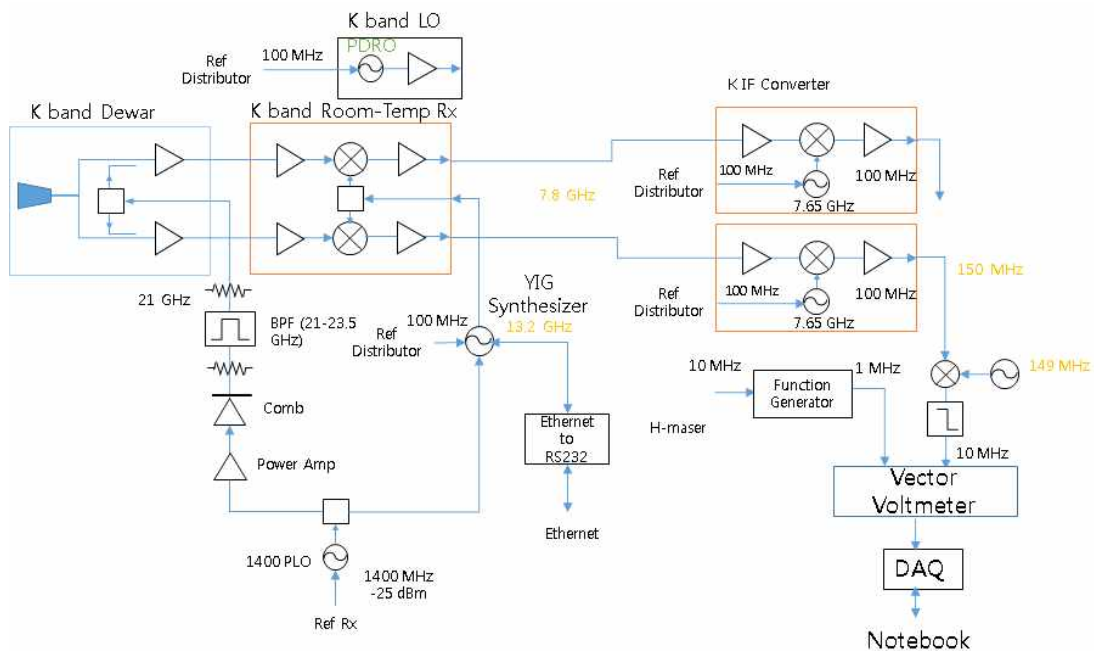
5. 세종 VLBI 시스템 적용을 위한 현장 성능 시험

가. 주파수 합성기 설치 및 수신기 위상 안정도 측정 setup 구성

제작된 주파수 합성기는 세종 망원경에서 현장 성능 시험에 사용될 것이다. 주파수 합성기의 주파수를 22GHz 수신기의 국부발진기 주파수인 13.25GHz 와 가장 인접한 13.2GHz에 설정하여 VLBI 관측에 사용되도록 할 것이다. 먼저 21GHz 위상 보정 신호를 발생시켜서 기존 국부발진기와 새로 설치한 주파수 합성기의 위상 안정도를 비교하였다. 그림 29에는 주파수 합성기를 연결한 상태에서의 위상 안정도 측정 setup을 보였다. 위상 보정 신호 인가를 위한 comb generator, 전력 증폭기 등의 연결은 그림 6-11에서처럼 기존 13.25GHz PDRO를 국부발진기로 사용했을 때의 위상 안정도 측정을 할 때와 같다.



<그림 6-28> 주파수 합성기를 22GHz 수신기 옆에 설치한 사진



<그림 6-29> 주파수 합성기를 22GHz 수신기의 국부발진기로 사용했을 때 위상 안정도 측정 구성

나. 망원경 성능 시험을 위한 프로그램 작성

온도 변화에 따른 위상 안정도 측정을 위해서 노트북 컴퓨터에서 파이썬을 활용한 프로그램을 사용하여 DAQ(Data Acquisition unit), Vector Voltmeter와 GPIB 통신을 수행하여 위상 안정도를 측정하였다. 수신기 실의 온도는 관측 시스템의 모니터 값을 별도로 저장하였다.

다. 개발된 주파수 합성기를 활용한 수신기 위상 안정도 측정

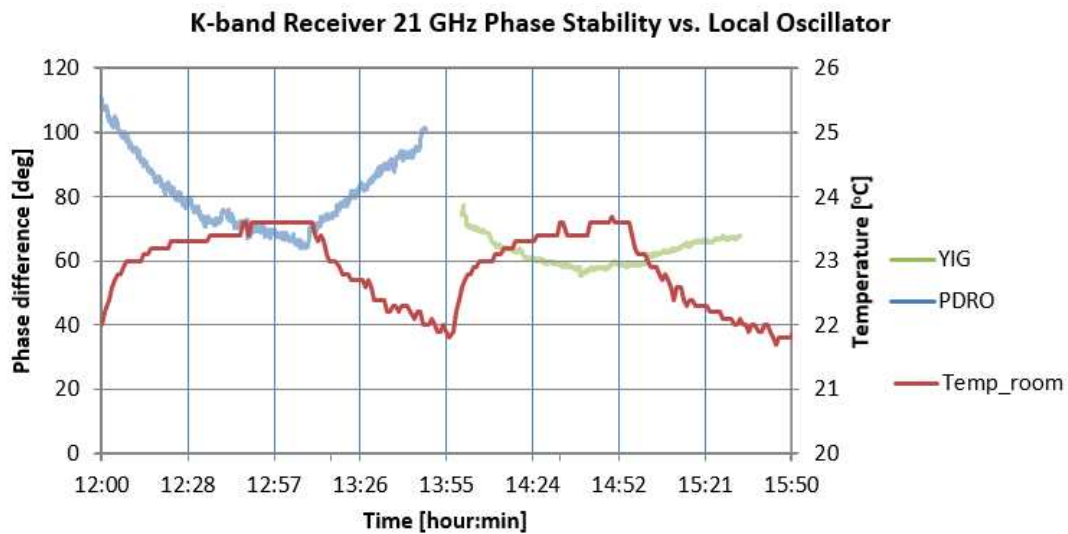
안테나의 고도를 0도→80도→50도→20도→0도 순서로 바꾸면서 측정을 하였다. 처음에는 기존의 PDRO 국부발진기 사용을 했을 때의 위상 안정도를 측정하였고, 다음에 같은 순서의 안테나 고도에 대해서 YIG 주파수 합성기를 국부발진기로 사용했을 때의 위상 안정도를 측정하였다. 표 6-11에 안테나 고도를 바꾸었을 때의 시간을 표시하였다. 그림 30에 국부발진기에 따른 위상 안정도 측정 결과를 보였다. 그림 30의 PDRO를 사용한 결과는 그림 13과 같은 그래프이다.

안테나의 고도 변화에 따라 안테나 수신기 실의 온도는 1.5도 정도 변화였다. 22GHz 수신기 국부발진기로 YIG 주파수 합성기를 사용할 때는 20 deg, PDRO를 국부발진기로 사용할 때는 40 deg 정도의 위상 변화를 보여, 2배 정도의 안정도 차이를 보였다. PDRO 13.25GHz 출력 주파수는 100MHz 기준 주파수에 비해 132.5 배의 주파수 차이가 있지만, YIG 주파수 합성기의 13.2GHz 출력 주파수는 1.4GHz 기준 주파수 신호의 10배가 되지 않는다. 기준 주파수와 출력 주파수의 차이가 작을수록 온도에 따른 위상 변화가 작게 나타난 것이다.

그림 8의 실험실에서 측정한 PDRO와 YIG 주파수 합성기의 안정도가 8 deg/°C와 2 deg/°C인 것을 고려하면, 그림 30의 측정 결과는 BBC(BaseBand Converter)의 7.65GHz 국부발진기, 100MHz 기준신호 발생기의 안정도의 영향이 포함된 것이다. 특히 기준신호 전송시스템에서 사용된 1.4GHz 기준신호를 직접 사용하지 않고, 100MHz 로 훨씬 낮은 기준 주파수 신호를 만든 후에 다시 높은 국부발진 주파수를 만드는 과정에 위상 안정도가 크기 나빠질 수 있는 것이다. 1.4GHz에서 100MHz를 생성하는 장치의 위상 안정도 측정은 별도로 하지 못하였다. 향후 100MHz 신호 발생기 자체의 위상 안정도를 측정하여 검증할 필요가 있다.

<표 6-11> 안테나 고도 변화 시간

안테나 고도 변화	안테나 고도 변경 시간[시:분]	
	PDRO 국부발진기 사용	YIG 주파수 합성기 사용
0도 → 80도	11:57	13:52
80도 → 50도	12:45	14:30
50도 → 20도	13:05	14:50
20도 → 0도	13:40	15:20



<그림 6-30> 안테나 고도와 수신기 실 온도 변화에 따른 22GHz 수신기의 위상 안정도 측정 결과

6. 22/43GHz 수신기 광대역화 구현을 위한 주파수 합성기 적용 방안 마련

기존에 설치되어 있던 13.25GHz PDRO(Phase-locked Dielectric Resonator Oscillator) 국부발진기를 사용한 경우에 비해 10-20GHz YIG 주파수 합성기를 사용한 경우에 21GHz에서의 수신기 위상 안정도가 2배 정도로 개선되었다. 하지만 YIG 주파수 합성기를 사용한 경우에도 안테나 고도 변화에 따라 20 deg 정도의 위상 변화가 나타났다. 이는 BBC 7.65GHz 국부발진기의 영향으로 추정된다. 즉, BBC 국부발진기도 기준신호 전송시스템의 1.4GHz 기준신호를 활용한 주파수 발생기를 사용해야 수 deg 이내의 수신기 안정도를 얻을 수 있을 것이다.

앞의 표 6-1과 6-2에서 IF 주파수에 따른 광대역 수신기의 국부발진 주파수를 정리하였다. 본 과제에서는 기존처럼 8-10GHz IF 주파수를 사용하는 경우에 맞춰서 10-20GHz의 넓은 범위의 주파수합성기를 제작하였지만, DBBC3의 4-15GHz의 넓은 IF 주파수 대역을 활용할 수 있다면, 광대역 22/43GHz 수신기의 국부발진기 주파수를 14, 18GHz의 2 개의 주파수만 사용하는 것이 시스템의 운용을 쉽게 할 수 있을 것이다.

향후 43GHz 수신기용 국부발진기, BBC용 국부발진기의 개선을 위해서는 1.4GHz 하모닉 신호 발생기를 공용으로 사용하고, YIG 발진기와 PLL 회로만 각 국부발진기용으로 사용할 수 있다. YIG 대신에 크기도 작고 가격도 저렴한 VCO(Voltage Controlled Oscillator)를 사용하는 방법이 있다. VCO는 YIG에 비해 위상 잡음 특성이 높지만, 1.4GHz 기준신호의 위상 잡음이 좋으면 높은 Loop bandwidth 설정으로 YIG 주파수합성기에 근접한 위상 잡음을 얻을 수 있다.

7. 결론

세종 우주측지관측센터는 측지 VLBI 관측을 위한 직경 22m 전파망원경과 수신기, 데이터 처리장치 및 VLBI 관측데이터를 상관처리 할 수 있는 DiFX 상관처리 서버를 구축·운영하고 있다. 2012년 국제기구인 IVS 가입 이후 정기적으로 측지 VLBI 관측에 참여하며 국내외 측지 VLBI 성과에 기여하고 있다. 하지만, 최근의 급격한 기술개발 및 발전으로 인하여 백엔드 시스템 및 상관처리 시스템의 고도화가 불가피하게 되었고, 이를 활용한 실시간 관측 및 상관처리(e-VLBI)와 국내외 VLBI 네트워크 협력이 더욱 중요하게 되었다. 또한 고도화된 백엔드 시스템을 지원할 수 있는 VLBI 데이터 분석 가속화를 위한 주파수 합성기 개발이 필요하게 되었다. 이를 위해 VLBI 데이터 분석 가속화 기술개발 연구가 수행되었으며, 그 결론은 다음과 같다.

1. 세종의 백엔드 시스템인 DBBC3를 기존 2L2H에서 4L4H로 업그레이드를 진행하고 최적화 작업을 완료하였다. 세종 DBBC3의 입력 주파수 대역에 맞춘 주파수 합성기를 개발하였다. 이로써 세종은 S/X(2/8GHz) 대역과 K/Q(22/43GHz) 대역에서 4개의 광대역 VLBI 데이터 획득이 가능하게 되었다.

2. 생성된 VLBI 데이터는 상관기를 이용하여 처리하는데, 신속한 상관처리를 위한 시간지연 오차 보정 및 데이터 저장·관리·처리를 자동화하는 절차를 정립하고 관련 유틸리티 프로그램을 개발하였다. 세종 전파망원경과 신규 컴퓨팅 노드 3기를 추가하여 성능이 향상된 리눅스 서버에 DiFX 소프트웨어 상관기를 설치하고 그 성능을 최적화 시켰다. 이로써 전파망원경 4기의 2Gbps e-VLBI 관측 및 상관처리 테스트를 성공적으로 수행하였다.

3. 향상된 VLBI 데이터 분석 가속화 기술을 기반으로 측지·천문, 시공간 융합연구 및 우주탐사 등 활용 가능한 연구 분야를 제시하였고, 실제로 세종 우주측지관측센터와 긴밀하게 협력하여 해당 연구들을 수행하고 있다. 특히, KVN을 비롯한 4개국 7개 기관이 참여하는 동아시아 VLBI 관측망(EAVN) 유관 기관들과 EAVN 공동 관측운영 및 협력에 관한 양해각서를 2021년 체결하고, 국내외 VLBI 네트워크 협력 기반을 구축하였다.

4. 국내외 VLBI 공동관측 협력을 위해 필수적인 세종 22m 전파망원경의 시스템 성능지표 산출하였다. 전파망원경 시스템의 성능지표는 VLBI 관측 협력을 위한 필수 지표로 국제적으로 통용되는 지표들을 산출하였으며, 세종 망원경의 우수한 성능을 확인할 수 있었다.

이상 ‘VLBI 데이터 분석 가속화 기술개발 연구[2차]’를 통하여 세종 우주측지관측센터의 VLBI 관측 시스템의 고도화와 함께 VLBI 활용 확대를 위한 국내외 협력 기반의 틀을 구축하였다. 하지만, 지속적으로 확대되고 있는 다양한 관측 요구조건들을 수용하기 위해서는 추가

적인 고도화 작업도 필요함을 제안하였다. 특히, 보다 넓은 IF 주파수 대역을 위한 시스템 고도화 및 43GHz 수신기 관련 국부발진기 성능 개선이 필요할 것으로 예상된다. 또한 DBBC3 입력 대역을 현재 4개에서 6개 또는 최대 8개의 입력 대역 확장도 고려할 수 있으며, 이에 따른 기록시스템 및 상관처리 시스템의 확장이 필요할 수 있다. 특히, 최근 그 중요성이 커지고 있는 고주파수 측지 VLBI 관측에서 독보적인 위상을 갖고 있는 세종 우주측지관측센터의 전과망원경은 지속적인 시스템 고도화 및 성능개선을 통하여 선도적인 측지 VLBI 관측국으로 자리매김 할 수 있을 것이다.

참고 문헌

- [1] Deller, Adam T., et al. "DiFX: a software correlator for very long baseline interferometry using multiprocessor computing environments." Publications of the Astronomical Society of the Pacific 119.853 (2007): 318.
- [2] Deller, A. T., et al. "DiFX-2: a more flexible, efficient, robust, and powerful software correlator." Publications of the Astronomical Society of the Pacific 123.901 (2011): 275.
- [3] Hobiger, Thomas, et al. "4-5 Real-time Determination of UT1 by Means of e-VLBI." Journal of the National Institute of Information and Communications Technology Vol 57.3/4 (2010).
- [4] Klotek, Grzegorz, et al. "Geodetic VLBI for precise orbit determination of Earth satellites: a simulation study." Journal of Geodesy 94 (2020): 1-26.
- [5] Weimin, Zheng, Chen Zhong, and Tong Li. "e-VLBI applications of Chinese VLBI network." Seventh General Meeting (GM2012) of the international VLBI Service for Geodesy and Astrometry (IVS. 2012.
- [6] Conway, J. E. "Scientific Applications of e-VLBI." Proceedings of the 8th International e-VLBI Workshop. 22-26 June 2009. Madrid, Spain. 2009.
- [7] KOYAMA, Yasuhiro, et al. "5-2 Research and Developments for e-VLBI Utilizing Global High Speed Network Connections." Journal of the National Institute of Information and Communications Technology Vol 52.3/4 (2005).
- [8] Thompson, A. R., et al., Interferometry and Synthesis in Radio Astronomy, 2nd Edition, p. 257

용어 및 약어 정리

- DiFX (Distributed FX-style software correlator): 소프트웨어 상관기
- FITS (Flexible Image Transport System): 천문관측에 사용되는 표준 파일 포맷 형식
- VERA (VLBI Exploration of Radio Astrometry): 일본 측성학 중점 VLBI 관측망
- CVN (Chinese VLBI Network): 중국 VLBI 관측망
- DBBC (Digital Base Band Converter): 디지털 기저대역 변환기
- RF (Radio Frequency): 전파(관측) 주파수
- IF (Intermediate Frequency): 중간 주파수
- DDC_U, DDC_L (Digital Down Converter, Upper, Lower):
디지털 주파수 변환기, 고주파/저주파 대역
- BW (Bandwidth): 대역폭
- VDIF (VLBI Data Interchange Format): VLBI 데이터 교환 포맷 (표준 데이터 포맷)
- SmpBit (Sampling Bit): 샘플링 비트
- Tsys (System noise temperature): 시스템 잡음 온도
- VFC (Voltage Frequency Converter): 전압 주파수 변환기
- LO (Local Oscillator): 국부발진기
- vex (VLBI experiment): VLBI 관측 파일 포맷
- SCHED (VLBI Schedule): VLBI 관측 스케줄 생성을 위한 프로그램
- OCTAD (Optically Connected Array for VLBI Exploration Analog-to-Digital
Converter): VLBI 관측용 아날로그-디지털 변환기
- SLURM: 리눅스 기반의 클러스터 스케줄 매니저 프로그램
- MTU (Maximum Transmission Unit): 네트워크 최대 전송 단위
- IVP (Invariant Point): 전파망원경의 수직축과 수평축이 만나는 고정점
- H-maser (Hydrogen maser): 수소원자시계
- PLL(Phase Locked Loop): 위상 동기(고정) 루프
- PLO(Phase Locked Oscillator): 위상 동기(고정) 발진기

- PDRO(Phase Locked Dielectric Resonator Oscillator): 위상 고정 유전체 공진 발진기
- Power divider: 파워(전압) 분배기
- USB(Upper Side Band), LSB(Lower Side Band): 상측파대, 하측파대
- BPF(Band Pass Filter): 대역 통과 필터
- YIG synthesizer: YIG (Yttrium-Iron-Garnet) 기반의 신호발생기
- HFP(High Pass Filter), LPF (Low Pass Filter): 고주파수, 저주파수 대역 통과 필터
- Down Converter: 저대역 주파수 변환기
- Mark6: VLBI 데이터 초고속 기록기
- Total Power Detector: 전파망원경에 인가되는 전파 신호의 세기를 검출하는 장치
- ANTAB: VLBI 전파망원경 성능에 따른 잡음 온도 정보가 담긴 파일 형식
- Comb generator: 주파수 빔 생성기
- Mixer: 주파수 혼합기
- Vector Voltmeter: 입력 신호의 위상/전압 측정 전압계
- VCO(Voltage Controlled Oscillator): 전압 제어 발진기

부록

1. DBBC3 업그레이드 성능 시험 결과서
 2. 세종 DiFX e-VLBI 매뉴얼
 3. 세종 전파망원경 시스템 성능 지표
 4. 주파수 합성기 프로그램
 5. 주파수 합성기 점검 매뉴얼
-

부록 1. DBBC3 업그레이드 성능 시험 결과서

System under Test, Start date, Engineer										no y-cable no mounting bar
HL 035 / 3KR 15.11.22 J. C.										
DBBC3 Check / Update / Test list for generic DBBC3 V2 (8IF)										
M. Wunderlich 29 July 2019 last update: Feb. 2, 2022										
Modifications, optimizations, procedures		Check	Done	Fail	Pass	Remarks				
Basic system:	Open DBBC3 and remove electronics stack, GCoMos and PC.		✓							
	Remove V3 power distributor board.		✓							
	Adjust height of lower supporting bars, add / modify upper front bar, fan improvement		✓		✓	will stay for 41				
	Remove 2 EL power supplies, PC power supply and cables.		✓							
	Prepare / check 4 EL connection cables 21 cm long, 0.75 mm²		✓							
	Adjust power supplies on the bench according to photo documentation.									
	Power Supply 3.3V nom. -5V nom. +12V nom. +5V nom.									
	PS1 +3.37 -5.04 +12.1 +5.10		✓		✓					
	PS2 +3.37 -5.02 +12.0 +5.11		✓		✓					
	PS3		✓							
	PS4		✓							
	Check V4 power distributor for bad solderings and loose sockets		✓							
	Bench test of power supplies + distributor + 2 ADB3/Core3 to check cable resistance.									
	ADB3/Core3 block 3.3V nom. -5V nom. +12V nom. +5V nom.									
	Block 1+2 +3.32 -5.04 +12.1 +5.02		✓		✓					
	Block 3+4 +3.32 -5.02 +12.0 +5.03		✓		✓					
	Block 5+6		✓							
	Block 7+8		✓							
	Insert V4 power distributor board.		✓			V3 for 415				
	Insert 4 EL and PC power supplies according to photo documentation.		✓							
	Change 2 fuses to 8A medium-acting.		✓		✓					
	Replace flat GCoT control cable with round one, route differently to prevent cross-talk.		✓		✓					

	Check RF Mon for spectrum and level.									
	Check attenuator settings.									
	Check preliminary TP readings.									
	Check synthesizer output for frequency and level.									
	Check frequency doubler + high pass for functionality and level.									
	Check for artefacts with synthesizer off.									
	Check down conversion for LO / HI according to photo documentation.									
	Adjust synthesizer level to minimize unwanted mixing products.									
	IF G:									
	Test GCoMo Ctrl board according to photo documentation									
	Check RF Out for spectrum and level.									
	Check RF Mon for spectrum and level.									
	Check attenuator settings.									
	Check preliminary TP readings.									
	Check synthesizer output for frequency and level.									
	Check frequency doubler + high pass for functionality and level.									
	Check for artefacts with synthesizer off.									
	Check down conversion for LO / HI according to photo documentation.									
	Adjust synthesizer level to minimize unwanted mixing products.									
	IF H:									
	Test GCoMo Ctrl board according to photo documentation									
	Check RF Out for spectrum and level.									
	Check RF Mon for spectrum and level.									
	Check attenuator settings.									
	Check preliminary TP readings.									
	Check synthesizer output for frequency and level.									
	Check frequency doubler + high pass for functionality and level.									
	Check for artefacts with synthesizer off.									
	Check down conversion for LO / HI according to photo documentation.									
	Adjust synthesizer level to minimize unwanted mixing products.									
	Mount all (8) GCoMos.		✓		✓					
	Check / modify for mechanical stability.		✓		✓					

	Add serial no./name on front, side, extra lab. rear power switch	✓	✓	
GcaT:	Check board for the latest modifications according to photo documentation. Terminate SysCik output on rear panel.	✓	✓	
FilaIn (V3):	Check board for the latest modifications according to photo documentation.	✓	✓	
FilaOut (V2):	Check board for the latest modifications according to photo documentation.	✓	✓	
GcoMo:	Replace 3x 4 GHz (12V) amplifiers with 6 GHz (5V) amplifiers. Check / upgrade frequency doublers.	✓	✓	
ADB3L:	Check board for the latest modifications according to photo documentation. When present, remove High-pass filter on phase adapters. Replace it with a bit of wire. On this occasion check the phase adapters with the network analyzer. Add 4 GHz Low-pass filter to phase adapters, check them for similarity Create serial no. and register boards in database. Update firmware in CPLD with latest version lock_281118.	✓ ✓ ✓ ✓ ✓	✓ ✓ ✓ ✓ ✓	3L3157-3L3160
Core3H V1:	Remove 6A regulators and replace them with 20A type according to photo documentation. Add the XADC functionality according to photo documentation. Remove small aluminium heat sink and replace it with large aluminium or ceramic type. Create serial no. and register boards in database.			
Core3H V2:	Modify power regulators according to photo documentation. Modify the XADC functionality according to photo documentation. Remove small aluminium heat sink and replace it with large aluminium or ceramic type. Create serial no. and register boards in database.			
				3H2134-3H2137
Control-PC:	Provide HD made from DBBC3 master. Check PCI backplane for jumpers and correct resistors. Check / provide PCI boards, adapters, flat cables Replace PCI-7200 with PCI-7396 incl. Flat cables when required Verify drivers for ADLINK etc. Configure COM ports, programmers, USB hubs etc.	✓ ✓ ✓ ✓ ✓ ✓	✓ ✓ ✓ ✓ ✓ ✓	

	Insert one functional block (ADB3L + Core3H) at a time.	✓	✓	
Block #1:	Using OCT, calibrate ADB3L according to documentation. Fill in data sheet. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.	✓ ✓ ✓	✓ ✓ ✓	3L3157+3H2134 coh 5
	Repeat that for all (8) blocks.			
Block #2:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.	✓ ✓ ✓	✓ ✓ ✓	3L3158+3H2134 coh 6
Block #3:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.	✓ ✓ ✓	✓ ✓ ✓	3L3159+3H2134 coh 7
Block #4:	Using OCT calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.	✓ ✓ ✓	✓ ✓ ✓	3L3160+3H2134 coh 8
Block #5:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.	/		
Block #6:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.			
Block #7:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels. Using OCT, check Core3H for correlation amplitudes.			
Block #8:	Using OCT, calibrate ADB3L according to documentation. Using OCT, check Core3H for sampler power levels.			

Single component tests:					
	Place system in test environment with 10MHz, 1pps, GPS, broadband noise 0-4 GHz, etc.	✓			
	Populate only GCaT, Filaln, Fila Out and one known good GcoMo	✓			
	Test GCaT entirely, sample clocks, 1pps, 10 Mhz etc.	✓	✓		
	Check levels of the GCaT Reset on/off functionality for every single clock line.	✓	✓		+3/-34dBm
	Put GcoMo under test with complete set of extension cables on the system.	✓			
	Have it open with the down conversion parts removed.	✓			
IF A:	Test GcoMo Ctrl board according to photo documentation	✓		✓	
	Check RF Out for spectrum and level, using broadband noise 0-4 GHz	✓		✓	
	Check RF Mon for spectrum and level, using broadband noise 0-4 GHz	✓		✓	
	Check attenuator settings.	✓		✓	
	Check preliminary TP readings.	✓		✓	
	Check synthesizer output for frequency and level.	✓		✓	12202743 Source 1
	Check frequency doubler + high pass for functionality and level.	✓		✓	COM 9
	Check for artefacts with synthesizer off.	✓		✓	
	Check down conversion for LO / HI according to photo documentation.	✓		✓	LO
	Adjust synthesizer level to minimize unwanted mixing products.	✓		✓	10dB
	Repeat that for all (8) GCoMos.				
IF B:	Test GcoMo Ctrl board according to photo documentation	✓		✓	
	Check RF Out for spectrum and level.	✓		✓	
	Check RF Mon for spectrum and level.	✓		✓	
	Check attenuator settings.	✓		✓	
	Check preliminary TP readings.	✓		✓	
	Check synthesizer output for frequency and level.	✓		✓	12202743 Source 2
	Check frequency doubler + high pass for functionality and level.	✓		✓	COM 9
	Check for artefacts with synthesizer off.	✓		✓	
	Check down conversion for LO / HI according to photo documentation.	✓		✓	LO
	Adjust synthesizer level to minimize unwanted mixing products.	✓		✓	10dB

IF C:	Test GcoMo Ctrl board according to photo documentation	✓		✓	
	Check RF Out for spectrum and level.	✓		✓	
	Check RF Mon for spectrum and level.	✓		✓	
	Check attenuator settings.	✓		✓	
	Check preliminary TP readings.	✓		✓	
	Check synthesizer output for frequency and level.	✓		✓	12202397 Source 1
	Check frequency doubler + high pass for functionality and level.	✓		✓	COM 10
	Check for artefacts with synthesizer off.	✓		✓	
IF D:	Check down conversion for LO / HI according to photo documentation.	✓		✓	LO
	Adjust synthesizer level to minimize unwanted mixing products.	✓		✓	10dB
	Test GcoMo Ctrl board according to photo documentation	✓		✓	
	Check RF Out for spectrum and level.	✓		✓	
	Check RF Mon for spectrum and level.	✓		✓	
	Check attenuator settings.	✓		✓	
	Check preliminary TP readings.	✓		✓	
	Check synthesizer output for frequency and level.	✓		✓	12202397 Source 2
IF E:	Check frequency doubler + high pass for functionality and level.	✓		✓	COM 10
	Check for artefacts with synthesizer off.	✓		✓	
	Check down conversion for LO / HI according to photo documentation.	✓		✓	LO
	Adjust synthesizer level to minimize unwanted mixing products.	✓		✓	10dB
IF F:	Test GcoMo Ctrl board according to photo documentation	✓		✓	
	Check RF Out for spectrum and level.	✓		✓	
	Check RF Mon for spectrum and level.	✓		✓	
	Check attenuator settings.	✓		✓	
	Check preliminary TP readings.	✓		✓	
	Check synthesizer output for frequency and level.	✓		✓	
	Check frequency doubler + high pass for functionality and level.	✓		✓	
	Check for artefacts with synthesizer off.	✓		✓	
IF G:	Check down conversion for LO / HI according to photo documentation.	✓		✓	
	Adjust synthesizer level to minimize unwanted mixing products.	✓		✓	

Using OCT , check Core3H for correlation amplitudes.				
Mount all (8) blocks together with Filaln, FilOut.		✓		✓
Check / modify for mechanical stability.		✓		✓
To check the entire power chain it is mandatory to re-check ALL the voltages in the completed system, switched ON, not configured.				
ADB3/Core3 block 3.3V nom. -5V nom. +12V nom. +5V nom.				
Block 1+2 +3.32 -5.04 +12.1 +5.02		✓		✓
Block 3+4 +3.32 -5.02 +12.0 +5.03		✓		✓
Block 5+6				
Block 7+8				
Do a TP calibration over all GCoMos using broadband noise to equalize levels.		✓		✓
Software tests:				
Load DSC , apply broadband noise 0-4 GHz, run for prolonged time, check for synch loss.		✓		✓
With DSC , perform zero baseline tests		✓		✓
Load DDC , apply broadband noise 0-4 GHz, run for prolonged time, check for synch loss.		✓		✓
With DDC , check the CCCM / ADB Reset stability.		✓		✓
With DDC , perform zero baseline tests		✓		✓
Load OCT , apply broadband noise 0-4 GHz, run for prolonged time, check for synch loss.		✓		✓
With OCT , check the CCCM / ADB Reset stability.		✓		✓
With OCT , perform zero baseline tests		✓		✓
Perform zero baseline tests with all lfs in down conversion mode		✓		✓

ADB3L for		Hc 035 / 3K2				
M. Wunderlich		28.11.22				
Sampler:		0	1	2	3	Serial.No
ADB3L #1	offset	131	109	122	128	3C3157 + 3H2134
	gain	139	122	120	121	
addr: F	delay	81	409	615	899	
ADB3L #2	offset	127	110	117	116	3C3158 + 3H2135
	gain	128	133	128	120	
addr: D	delay	210	470	554	872	
ADB3L #3	offset	130	107	126	110	3C3159 + 3H2136
	gain	133	128	121	134	
addr: B	delay	45	485	539	865	
ADB3L #4	offset	126	120	118	122	3C3160 + 3H2137
	gain	135	112	128	139	
addr: 9	delay	198	490	537	960	

○ 평가 항목 및 내용 요약

[하드웨어 점검 및 평가]

- DBBC3 케이블, 구조, 보드, 구조체 지지대 등 점검
- 파워서플라이 업그레이드에 따른 전압 분포 수치
- 파워 스위치 안정도
- 4GHz (12V) 앰프 3개를 6 GHz (5V) 앰프로 교체
- 고대역 필터 및 저대역 필터 점검
- 펌웨어 업그레이드
- 입력 파워조절기 점검 및 발열을 위해 넓은 방열판 설치
- COM 포트 및 USB 허브 설정
- 1pps, 10MHz 기준 신호 테스트
- 관련 케이블 교체
- CcoMo 컨트롤 보드 테스트
- RF 출력, 모니터링, 감쇄기, 신디사이저 출력 및 레벨 등 확인
- OCT 모드 테스트 (Core3H, ADB3L 샘플러 파워 레벨, 상관계수 등)
- 전체 입력 파워(전력) 분포 및 스위치 점검
- 최종 기기성능 안정도 점검

[소프트웨어 점검 및 평가]

- DSC/DDC/OCT 로딩 및 동기화 손실 점검
- 최종 Zero Baseline 성능 테스트

3KR back before upgrade



3KR back after upgrade



3KR front before upgrade



3KR front after upgrade



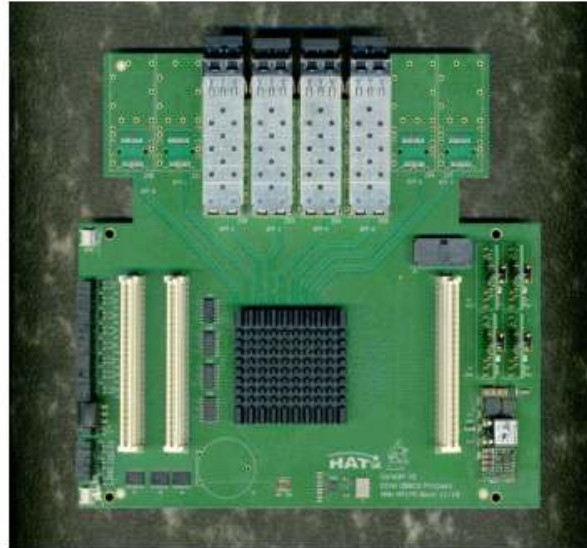
3KR top before upgrade



3KR top after upgrade



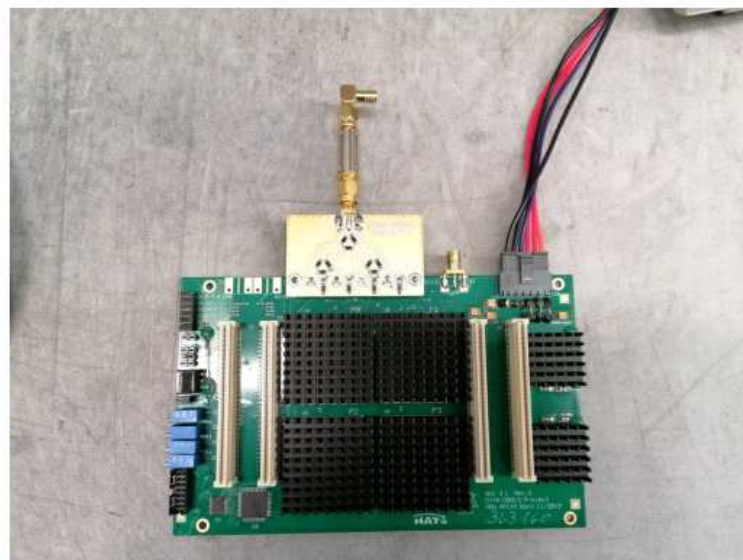
Core3H before upgrade



Core3H after upgrade



ADB3L after upgrade



부록 2. 세종 DiFX e-VLBI 매뉴얼

세종 DiFX 서버를 활용한 e-VLBI 매뉴얼

요약

이 보고서는 e-VLBI를 위한 세종의 DiFX 서버 (리눅스 클러스터)에 대한 점검, 벤치마킹 및 설정 결과를 설명한다.

리눅스 클러스터 시스템인 **Sejong belbi2**는 축지 VLBI 관측 데이터를 상관처리하는데 사용되고 있으며, 보다 높은 성능 향상 및 상관처리를 위하여 클러스터 노드가 추가되었다. DiFX는 전 세계에서 가장 널리 사용되고 있는 상관처리 소프트웨어로 **Adam Deller[1][2]**에 의해 개발되었고, 미국국립전파천문대(NRAO)를 비롯하여, 세종을 포함한 IVS 상관처리 센터 등에서 이용되고 있다.

2020년 VLBI 데이터 분석 가속화 기술개발 연구(1차년)에서 세종 리눅스 클러스터를 이용하여 과학기술연구망(Kreonet) 10Gbps 링크를 기반으로 한 테스트가 진행되었다. 실시간 e-VLBI 테스트 관측이 **Sejong**과 **KVN** 사이에서 1Gbps 및 2Gbps로 스트리밍된 4개 스테이션의 백엔드 데이터가 세종 belbi2 리눅스 클러스터에서 실시간 e-VLBI 상관처리로 시행되었다. 당시 세종 belbi2 리눅스 클러스터를 이용하여 1Gbps e-VLBI는 성공적으로 진행되었으나, 2Gbps e-VLBI의 경우는 세종 belbi2 리눅스 클러스터의 성능 한계로 인하여 안정적인 상관처리가 불가능하였다.

이 후 VLBI 데이터 분석 가속화 기술개발 연구(2차년)에서 세종 belbi2 DiFX 클러스터를 업그레이드를 진행하였고, 클러스터 헤드 노드 뒤에 있는 6개의 컴퓨팅 노드(기존 3개)를 사용할 수 있게 되었다. 해당 클러스터 상호 연결 및 Kreonet 연결은 이전과 같이 10G로 유지되며 최대 처리량은 ~9Gbps로 확인되었다. 따라서 업그레이드 된 세종 belbi2 DiFX 클러스터를 이용하면, 세종 1기와 KVN 3기에서 산출되는 각 2Gbps 데이터(총 2Gbps x 4기 = 8Gbps)를 실시간으로 처리할 수 있다.

최종 목표는 업그레이드된 세종 클러스터를 벤치마킹하고 튜닝하는 것으로, 세종 및 KVN 스테이션의 Mark6 기록기에서 합성 데이터 스트림을 사용하여 e-VLBI 성능을 확인하였다. 각 망원경의 Mark6 기록기로부터 업그레이드된 세종 클러스터로 스트리밍된 VDIF 데이터는 다수의 10분 정도의 긴 스캔에서도 패킷 손실 없이 망원경 4기 x 2Gbps(총 8Gbps)의 속도로 안정적으로 처리되었다.

최적의 설정을 위해 belbi2에서 iptables 기반 트래픽 포워딩을 사용하였다. Iptables 규칙은 들어오는 4개의 VDIF 스트림을 클러스터에서 각기 다른 노드로 전달하도록 설정함으로써, belbi2의 DiFX 성능 병목 현상을 방지할 수 있었고, 성공적으로 2Gbps e-VLBI를 위한 기반이 구축되었음을 확인하였다.

목차

1. 세종 DiFX 서버(belbi2) 설정	3
1.1 네트워크 인터페이스 설정	3
1.2 SSH 사용자 설정	3
1.3 e-VLBI를 위한 belbi2 방화벽 설정	4
1.4 OpenMPI를 위한 belbi2 방화벽 설정	4
1.5 DiFX 멀티캐스트 메시지(Multicast Messages)에 대한 방화벽 설정	6
1.6 관측국 4기 x 2Gbps 상과처리를 위한 데이터 스트림 리다이렉션을 위한 방화벽 설정	7
1.7 관측국 4기 x 2 Gbps e-VLBI 상과처리	8
1.8 SLURM 설정	9
1.9 DiFX 설정	9
1.10 Mark6 기록기의 "MTU" 경로 설정 문제	10
2. 성능 측정	11
3. 실시간 e-VLBI 상과처리	13
관측국 3기 x 1024 Mbps 테스트	13
관측국 4기 x 1024 Mbps 테스트	14
관측국 3기 x 2048 Mbps 테스트	14
관측국 4기 x 2048 Mbps 테스트	14

참고문헌

1. 세종 DiFX 서버(belbi2) 설정

우선 예전에 구축된 클러스터 노드(n1-n3)와 새 노드(n4-n6) 및 헤드(Head) 노드 belbi2에서 설정을 확인한다. Belbi2 리눅스 클러스터는 사전에 구축이 되었으며, 이전 belbi를 10Gbps Kreonet 망에 연결된 헤드 노드로 대체하였으며, 추가적인 변경 사항은 다음에서 설명한다.

1.1 네트워크 인터페이스 설정

최초 10Gbps 네트워크 인터페이스는 MTU 설정이 1500으로 되어있었다. 벤치마킹 후 성능 향상을 위하여 MTU를 9000으로 향상시켰다.

```
belbi2# ifconfig p3p1 mtu 9000
belbi2# pdsh -w n[1-6] ifconfig enp175s0f0 mtu 9000
```

참고로, 전체 클러스터의 10Gbps 네트워크 카드의 MTU 설정은 1500으로 되어있으나, 전체 클러스터의 10Gbps 네트워크 인터페이스에 대한 MTU를 9000으로 고정하는 것도 고려할 수 있다. 다른 방법으로는 현재의 설정에서 서브넷 설정을 조정하는 것도 고려할 수 있다. belbi2(p3p1 210.110.232.5/25)의 서브넷 구성과 클러스터 노드 n1-n6(enp175s0f0 210.110.232.xx/24; xx=21..26)의 더 넓은(광대역) 서브넷 구성에 약간의 차이가 있지만, 라우팅은 잘 작동하므로 belbi2에서 서브넷 설정을 수정하지 않았다.

1.2 SSH 사용자 설정

리눅스 클러스터의 병렬 컴퓨팅 계산을 위해 사용하는 OpenMPI는 모든 노드의 포트 22번을 기본 SSH 통신을 위해 사용하지만, belbi2의 경우는 포트 7171을 사용한다. 따라서 SSH 구성 파일의 설정위해 다음과 같이 활성화 하였다.

```
<<n1...n6:/home/difx/.ssh/config>>
# Added by JanW 2020-12-05 so that OpenMPI mpirun is be able to
# launch processes on belbi2 (n1-n6 have usual SSH port 22, belbi(2) has port
7171)
Host 210.110.232.5
    Port 7171
Host belbi2
    Port 7171
```

이론적으로 belbi2의 영구적인 방화벽 규칙을 사용할 수 있다. 노드 n1-6에서 시작되는 데이터 수신 트래픽을 처리하기 위하여 해당 TCP 포트 22번의 레이블을 TCP 포트 7171로 다시 지정해야 하지만, DiFX 구동 이외에 Belbi2의 기타 용도로 활용한 경우를 고려하여

실제로는 방화벽 규칙을 영구적으로 적용하지는 않았다. 따라서 DiFX를 구동할 때, 위의 SSH 사용자 구성을 활성화된 상태로 유지시켜야 한다.

```
belbi2# yum install iwpmc rdma-core
belbi2# pdsh -w n[1-6] yum -y install iwpmc rdma-core

belbi2# yum install librdmacm compat-opensm-libs librdmacm-utils
belbi2# pdsh -w n[1-6] yum -y install librdmacm compat-opensm-libs
librdmacm-utils
```

1.3 e-VLBI를 위한 belbi2 방화벽 설정

belbi2의 방화벽은 `rt_evibi` 서비스 설정(위치: `/etc/firewalld/services/rt_evibi.xml`)을 포함하여 다음과 같이 설정하였다.

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>Real-time eVLBI</short>
  <description>Real-time eVLBI testing UDP/IP from KVN Mark6s sending to
ports 12000-12003, 13000-13003</description>
  <port protocol="udp" port="12000-12003"/>
  <port protocol="udp" port="13000-13003"/>
  <port protocol="udp" port="14000-14003"/>
  <port protocol="udp" port="15000-15003"/>
  <protocol value="udp"/>
  <destination ipv4="210.110.232.5/24"/>
</service>
```

실시간 e-VLBI 상관처리를 위해 belbi2 헤드 노드는 Sejong 및 KVN 스테이션 백엔드에서 데이터 스트림을 수신하는 10Gbps UDP/IP VDIF 트래픽 입력 노드로 사용할 수 있도록 설정해야 한다. 이를 위해서는 DiFX가 belbi2에서도 실행되어야 한다. 이를 위해 SLURM(클러스터 컴퓨팅을 위한 스케줄러)은 "belbi2.n1-n6"로 할당하고, 모든 호스트는 MPI 관련 TCP 트래픽을 허용해야 한다.

1.4 OpenMPI를 위한 belbi2 방화벽 설정

DiFX 헤드 노드와 클러스터 노드 간의 MPI 통신을 허용하기 위해 belbi2가 클러스터 노드 n1-n6의 모든 10G 트래픽을 허용하도록 새로운 영구 방화벽 규칙을 다음과 같이 추가하였다.

```
belbi2#
firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.21 accept' --permanent
```

```

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.22 accept' --permanent

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.23 accept' --permanent

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.24 accept' --permanent

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.25 accept' --permanent

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" source
address=210.110.232.26 accept' --permanent

```

참고 1: 방화벽 규칙이 설정되어 있지 않으면 belbi2와 노드 간의 non-SSH 트래픽이 차단됩니다. 이 경우 OpenMPI mpirun의 오류 메시지는 "[n1][39270,1],1] [btl_tcp_endpoint.c:818:mca_btl_tcp_endpoint_complete_connect] connect() to 210.110.232.5 failed: No route to host (113)"와 같이 발생한다. 이 경우에는 위의 방화벽 규칙을 belbi2에 다시 적용해야 한다.

참고 2: 방화벽 규칙이 광범위하게 열려있다고 판단되면 firewall-cmd 명령에서 "--add-port=<first_port>-<last_port>/tcp"를 사용하여 포트를 원하는 범위로 제한 할 수 있다. 또한 이 stackoverflow 게시물에서 제안한 것처럼 openmpi-mca-params.conf 파일을 수정하여 제한할 수 있다.

stackoverflow:
<https://stackoverflow.com/questions/32748628/set-mpi-to-run-on-a-range-of-ports> 참조

You can limit the port range in the [openmpi-mca-params.conf](#) (from that link:)
 By default, two files are searched (in order):
 \$HOME/.openmpi/mca-params.conf: The user-supplied set of values takes the highest precedence.
 \$prefix/etc/openmpi-mca-params.conf: The system-supplied set of values has a lower precedence.

The following settings appear to be relevant for the used ports:

```

MCA btl: parameter "btl_tcp_port_min_v4" (current value:
"1024", data source: default, level: 2 user/detail,
type: int)
The minimum port where the TCP BTL will try to bind
(default 1024)
MCA btl: parameter "btl_tcp_port_range_v4" (current value:
"64511", data source: default, level: 2
user/detail, type: int)
The number of ports where the TCP BTL will try to
bind (default 64511). This parameter together with
the port min, define a range of ports where Open
MPI will open sockets.

MCA oob: parameter "oob_tcp_static_ipv4_ports" (current
value: "", data source: default, level: 9 dev/all,
type: string)
Static ports for daemons and procs (IPv4)
MCA oob: parameter "oob_tcp_dynamic_ipv4_ports" (current
value: "", data source: default, level: 9 dev/all,
type: string)
Range of ports to be dynamically used by daemons
and procs (IPv4)

```

See `ompi_info --param btl tcp --level 9` and `ompi_info --param oob tcp --level 9`. Setting these values should allow you to limit the port range that OpenMPI will attempt to use. But I am not quite sure how this would improve your security? Most attackers would be able to find any open port range? It usually is more important, what is actually listening on those ports, isn't it? Better move your firewall outside your cluster and make be sure of which services are listening on the open ports.

1.5 DiFX 멀티캐스트 메시지(Multicast Messages)에 대한 방화벽 설정

DiFX 소프트웨어는 상관처리를 수행하고, 발생하는 오류를 확인하고, 데이터에 대한 가중치(weighting)를 부여하고, 재생(playback) 장치의 상태 등에 관한 내용을 멀티캐스트 메시지(multicast message)로 출력한다. 이러한 메시지는 일반적으로 DiFX 유틸리티 `errormon2`, `statemon`, `mk6mon`, `mk5mon`를 통해서 모니터링 할 수 있다. 헤드 노드에서 이러한 메시지를 수신하려면 1GbE 및 10GbE 인터페이스의 방화벽에서 멀티캐스트 트래픽을 다음과 같이 허용해야 한다.

```

belbi2#
firewall-cmd --permanent --zone=external --add-rich-rule="rule family=ipv4
destination address=224.0.0.0/4 protocol value=udp accept"
firewall-cmd --permanent --zone=trusted --add-rich-rule="rule family=ipv4
destination address=224.0.0.0/4 protocol value=udp accept"

```

DiFX 메시지 그룹에 대한 멀티캐스트는 다음과 같이 리눅스 **omping** 유틸리티를 이용하여 테스트할 수 있다.

```
belbi2# yum install omping
belbi2# pssh -w n[1-6] yum -y install omping

difx@belbi2 DiFX-2.5.2 ~> omping -m 224.2.2.1 -p 50201 belbi2-p3p1
n1-enp175s0f0.cluster
difx@n1 DiFX-2.5.2 ~> omping -m 224.2.2.1 -p 50201 belbi2-p3p1
n1-enp175s0f0.cluster
05:03:37.253932 IP belbi2-p3p1.50201 > n1-enp175s0f0.cluster.50201: UDP,
length 45
05:03:37.254938 IP belbi2-p3p1 > igmp.mcast.net: igmp v3 report, 1 group
record(s)
05:03:38.150966 IP belbi2-p3p1 > igmp.mcast.net: igmp v3 report, 1 group
record(s)
05:03:38.255084 IP belbi2-p3p1.50201 > n1-enp175s0f0.cluster.50201: UDP,
length 45
05:03:38.296203 IP n1-enp175s0f0.cluster.50201 > belbi2-p3p1.50201: UDP,
length 45
05:03:38.296349 IP belbi2-p3p1.50201 > n1-enp175s0f0.cluster.50201: UDP,
length 64
...
```

1.6 관측국 4기 x 2Gbps 상과처리를 위한 데이터 스트림 리다이렉션을 위한 방화벽 설정

원칙적으로 Kreonet 10Gbps 링크는 세종 리눅스 클러스터에 총 8~9Gbps 연결 성능을 지원한다.

성능 테스트를 통하여 belbi2 서버 자체의 DiFX만으로는 최대 6Gbps에 해당하는 “2~3기 관측국 x 2Gbps”에 대해 안정적으로 처리할 수 있음을 확인하였다. 세종 belbi2 서버가 belbi2 단독으로 스트리밍된 관측국 4기의 2Gbps 데이터(총 ~8.2Gbps)를 수신하여 DiFX로 상과처리하기에는 시스템 부하가 너무 높다는 것을 확인하였다.

하지만, belbi2와 n1-6은 10Gbps Kreonet에 연결된 동일한 10G 네트워크 스위치에 물려있기 때문에 각 관측국(station)에서 들어오는 데이터 스트림을 다른 노드로 리다이렉션하여 belbi2에서 발생하는 막간의 부하를 줄여줌으로써, 관측국 4기 x 2Gbps 데이터 스트림에 대한 실시간 e-VLBI를 상과처리를 진행할 있다. 이를 위해 리눅스 클러스터에서는 다음과 같은 두 가지 방법을 고려할 수 있다.

1. Kreonet에서 belbi2 이 외에 하나 이상의 클러스터 노드에 대한 외부 액세스를 허용하고 이에 따라 각 망원경(관측국)의 백엔드(e.g., Mark6)에서 출력되는 데이터 스트림의 IP를 조정하거나,

2. Belbi2에서 일부 관측국의 데이터를 클러스터의 다른 노드로 전달하는 기능 추가하여 iptables/방화벽 규칙으로 구성할 수 있다. (이때, 각 관측국 백엔드의 데이터 스트림 IP는 여전히 belbi2로 설정)

이 중, 두 번째 옵션의 경우 다음과 같이 임시 방화벽 규칙을 belbi2에 적용하여 관측국 4기 x 2Gbps 실시간 e-VLBI 상관처리 성능 테스트를 수행하였다.

```
belbi2#
firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" forward-port
port=12000 protocol="udp" to-port=12000 to-addr="210.110.232.21"'

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" forward-port
port=12001 protocol="udp" to-port=12001 to-addr="210.110.232.22"'

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" forward-port
port=12002 protocol="udp" to-port=12002 to-addr="210.110.232.23"'

firewall-cmd --zone=external --add-rich-rule 'rule family="ipv4" forward-port
port=12003 protocol="udp" to-port=12003 to-addr="210.110.232.24"'

firewall-cmd --list-all
```

포트 12000(KVN 탐라안테나 데이터 스트림)에서 들어오는 UDP는 노드 n1로, 포트 12001(KVN 울산안테나 데이터 스트림)은 노드 n2로, 포트 12002(KVN 연세안테나 데이터 스트림)은 노드 n3으로, 포트 12003(세종우주측지센터 안테나 데이터 스트림)은 노드 n4로 전달된다.

새 방화벽 규칙을 적용하기 전에 반드시 활성화(active)된 수신 UDP 데이터 스트림을 모두 중지해야 한다. 규칙은 "새로운(new)" 데이터 스트림에만 적용되기 때문이다. 참고로 이 방화벽 규칙을 적용하는 동안 여전히 열려 있는 스트림은 기존 스트림을 중지했다가 다시 시작할 때까지 리디렉션되지 않는다.

위에서 언급한 방화벽 규칙을 중지시키고 원래대로 돌아가려면 다음 명령을 입력한다.

```
belbi2# firewall-cmd --reload
```

1.7 관측국 4기 x 2 Gbps e-VLBI 상관처리

DiFX의 .v2d 파일의 안테나 DATASTREAM은 어떤 관측국(망원경)의 어떤 데이터 스트림이 어느 클러스터 노드로 전송되어야 하는지를 기술한다. 이를 위해서 'machine=<hostname>' 항목을 사용할 수 있다. 앞서 기술한 방화벽 규칙을 고려한 datastream 설정은 다음과 같다.

```
ANTENNA Kt { datastreams = dsKt }
ANTENNA Ku { datastreams = dsKu }
ANTENNA Ky { datastreams = dsKy }
ANTENNA Kv { datastreams = dsKv }
```

```

DATASTREAM dsKt { networkPort = 12000   UDP_MTU = 8260   machine = n1 }
DATASTREAM dsKu { networkPort = 12001   UDP_MTU = 8260   machine = n2 }
DATASTREAM dsKy { networkPort = 12002   UDP_MTU = 8260   machine = n3 }
DATASTREAM dsKv { networkPort = 12003   UDP_MTU = 8260   machine = n4 }

```

1.8 SLURM 설정

세종 리눅스 클러스터에는 이미 SLURM이 구성되어있고, 새로 추가된 노드를 e-VLBI를 위해 만든 특별 SLURM 파티션 "evlbi"에 추가하여 설정하였다.

7노드 304코어 "evlbi" 파티션(belbi2 및 n1-6)에서 벤치마크 프로그램 또는 DiFX e-VLBI 상관처리를 실행하기 위하여 다음과 같이 'salloc'을 사용하여 셸(shell)과 자원 할당(allocation)을 할 수 있다.

```

difx@belbi2 DiFX-2.5.2 ~> sinfo -o%C
CPUS(A/I/O/T)
0/304/0/304
difx@belbi2 DiFX-2.5.2 ~> salloc -p evlbi --ntasks=304 --nodes=7

difx@belbi2 DiFX-2.5.2 ~> cd /home/difx/evlbi/sejong2020difx/mpiBench
difx@belbi2 DiFX-2.5.2 mpiBench> mpirun -np 7 -H
belbi2-p3p1,n1-enp175s0f0,n2-enp175s0f0,n3-enp175s0f0,n4-enp175s0f0,n5-enp175
s0f0,n6-enp175s0f0 $abspath/mpiBench_sj2 -b 32k -e 64M -t 10000000 Alltoall

```

1.9 DiFX 설정

e-VLBI DiFX 상관처리를 위하여 salloc될 할당하거나 sbatch 배치 작업을 진행하기 위한 환경 설정은 다음과 같다.

```

export DIFX_MPIRUNOPTIONS="--report-bindings --mca btl_tcp_if_exclude
lo,eml,eth0 --mca mpi_yield_when_idle 1 --mca rmaps seq --map-by node
--bind-to none"

```

위 설정에서 'tcp_if_exclude'는 모든 통신이 10Gbps 네트워크 인터페이스를 통해 이루어지도록 하기 위하여 1Gbps 네트워크 인터페이스를 피하도록 설정한다. 마지막 부분에서 '--bind-to none' 옵션 대신 '--bind-to numa'를 사용하여 다른 코어로 마이그레이션하는 대신 각 프로세스를 특정 코어에 바인딩시킬 수 있다. 실제 테스트 결과 numa 설정에 따른 DiFX 상관처리 성능은 큰 차이가 없는 것으로 여겨진다.

기존에 각 관측국에서 전송된 파일을 모두 저장한 후, DiFX 상관처리를 수행하는 경우에는 DiFX/OpenMPI .machines 파일을 DiFX genmachines로 생성한 후 상관처리를 진행한다. 하지만, 실시간 e-VLBI의 경우에는 vex2difx에서 생성한 파일을 사용해야 한다. e-VLBI의

경우 DiFX가 startdifx 명령을 통해 시작되면 genmachine이 vex2difx가 생성한 컴퓨터 파일에 덮어쓰지 않도록 매개 변수 "startdifx --nomachines" 옵션을 추가해야 한다.

e-VLBI의 경우, UDP를 통해 VDIF를 수신하는 호스트는 v2d 데이터 스트림 섹션에 지정되어야 한다. 또한 컴퓨팅 노드는 v2d 파일의 전역 섹션(global section)에 지정되어야 한다. 예를 들어:

```
machines = belbi2, n1,n1,n1,n1,n1,n1,n1,n1,n1,n1,n1,n1,n1,
            n2,n2,n2,n2,n2,n2,n2,n2,n2,n2,n2,n2,n2,
            n3,n3,n3,n3,n3,n3,n3,n3,n3,n3,n3,n3,n3,
            n4,n4,n4,n4,n4,n4,n4,n4,n4,n4,n4,n4,n4,
nThread = 1
nCore = 53
```

또는 관측국 4기 x 2048Mbps의 경우 가장 성능이 잘 나오는 노드 설정은 다음과 같다.

```
machines = belbi2, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6,
            n1,n2,n3,n4,n5,n6
nThread = 4
nCore = 53
```

참고로 machines=...의 각 목록 항목은 하나의 DiFX mpifxcorr 프로세스에 해당한다.

이상으로 Belbi2를 이용한 관측국 4기 x 2Gbps 실시간 e-VLBI 상관처리에 대한 방화벽 설정 및 DiFX 상관처리 설정에 대해 기술하였다.

1.10 Mark6 기록기의 “MTU” 경로 설정 문제

세종 belbi2 리눅스 클러스터를 이용한 DiFX e-VLBI 상관처리 테스트 결과에 따르면 KVN 관측국(203.250.156.xxx)에서 belbi2와 Mark6 10Gbps 네트워크 인터페이스 사이의 네트워크 경로 MTU는 이상적인 9000바이트로 원활하게 통신이 되었다.

데이터를 수신하는 곳에서(예: DiFX, vdfitimeUDP) 좋은 성능을 발휘하려면 큰 MTU 인터페이스가 필요하다. 이러한 데이터 수신은 서버의 CPU/커널 성능으로 인하여 초당 특정 패킷 속도로 제한된다. VLBI 백엔드 시스템의 데이터 스트리밍 소프트웨어는 8192바이트(8224바이트 UDP)의 VDIF 프레임 크기를 사용한다. 이는 9000의 MTU뿐만 아니라 모든 2의 거듭제곱에 해당되는 VLBI 데이터 스트리밍 속도에 적합하며, 이를 통해 UDP 스트리밍 당 최대 8Gbps의 속도로 데이터를 수신할 수 있다.

그러나 종종 테스트 중에 데이터가 KVN 관측국(예: vdfistream 프로그램)의 Mark6에서 전송되어 오는 경우, 아래와 같이 Mark6의 belbi2에 대한 경로 MTU가 1500으로 감소하는 현상을 보였다.

```
oper@Mark6-US-1:~$ tracepath 210.110.232.5
```



```
1: Mark6-US-1.kreonet.re.kr 0.047ms pmtu 1500
2: ...
```

이에 따라서 세종 belbi2 DiFX 서버에서는 상관처리 진행률 리포트에서 80% 미만의 불량한 데이터 가중치가 표시되었고, vdiffimeUDP에서는 패킷 손실이 크게 나타났음을 알 수 있었다.

이처럼 KVN 관측 데이터 스트림의 네트워크 인터페이스 상에서의 MTU 9000 설정이 1500으로 다운그레이드되는 원인은 명확하지 않지만, 이런 현상 발생은 Mark6 기록기에서만 발생했고, Kreonet에 직접 연결된 다른 백엔드 시스템 – 예를 들어, KVN에서 사용하고 있는 OCTAD나 세종의 DBBC3 또는 KVN의 ADS-1000/FILA10G 시스템 –에서는 발생하지 않아야 한다.

이렇게 간헐적으로 발생하는 MTU 다운그레이드 현상을 일시적으로 수정하려면 먼저 Mark6에서 전송하고 있는 데이터 스트리밍을 취소하고 다음과 같이 해당 Mark6에서 경로 캐시를 플러시(flush)시킨 다음에 데이터 스트리밍을 다시 시작해야 한다.

```
root@Mark6-US-1:~# echo 1 > /proc/sys/net/ipv4/route/flush
oper@Mark6-US-1:~$ tracepath 210.110.232.5
1: Mark6-US-1.kreonet.re.kr 0.051ms pmtu 9000
2: ...
```

2. 성능 측정

10GbE를 통한 클러스터 내 데이터 전송은 모든 노드가 10G 스위치를 통해 다른 모든 노드로 데이터를 전송하는 모드에서 mpiBench를 사용하여 그 성능을 테스트 하였다. 이를 위한 설정은 다음과 같다.

```
$ export MPIBENCH_PROG=/home/difx/jwagner/sejong2020difx/mpiBench/mpiBench
$ export OMPI_MCA_coll="tuned"
$ export OMPI_MCA_mtl="psm"
$ export OMPI_MCA_btl=tcp,self,sm
$ export OMPI_MCA_oob=tcp
$ export OMPI_MCA_btl_tcp_if_exclude=lo,eth0,em1 # excludes 1GbE iface,
keeps belbi2 p3p1 and node eth2 10G ifaces
```

그리고 belbi2와 n1-3(호스트 4개), belbi2와 n1-6(호스트 6개) 간의 테스트 설정은 다음과 같다.

```
$ mpirun -np 4 -H belbi2-p3p1,n1-enp175s0f0,n2-enp175s0f0,n3-enp175s0f0
$MPIBENCH_PROG -b 256k -e 64M -t 10000000 Alltoall | tee out.txt

$ mpirun -np 7 -H
belbi2-p3p1,n1-enp175s0f0,n2-enp175s0f0,n3-enp175s0f0,n4-enp175s0f0,n5-enp175
s0f0,n6-enp175s0f0 $MPIBENCH_PROG -b 256k -e 64M -t 10000000 Alltoall | tee
out.txt
```

성능측정 결과 수 ~100MB/s 정도의 차이가 발생했다. 세종 DiFX 성능을 업그레이드 진행 전의 3노드 클러스터와 확장된 6노드 클러스터 및 10Gbps 인터페이스가 있는 확장된 6노드 클러스터에서 e-VLBI 트래픽 설정에 따라 MTU를 1500과 9000로 설정하여 테스트한 결과는 다음과 같다.

MPI Msg Size	Per-host rate (MB/s) 4 hosts: belbi2, n1-3 mtu 1500	Per-host. rate (MB/s) 7 nodes: belbi2, n1-6 mtu 1500	Per-host rate (MB/s) 7 nodes: belbi2, n1-6 mtu 9000
32 kB	673.429	960.150	1088.433
64 kB	757.638	966.922	848.675
128 kB	977.571	934.303	915.573
256 kB	1198.286	1407.242	1071.699
512 kB	1431.238	1568.577	1344.174
1 MB	1564.454	1671.023	1720.343
2 MB	1577.999	1462.780	1012.316
4 MB	1512.070	1447.408	1021.371
8 MB	1620.021	1518.103	1069.911
16 MB	1853.681	1742.082	1237.450
32 MB	1921.193	1786.694	1363.823
64 MB	1951.620	1820.052	1633.729

위 결과는 각 실행마다 다르지만 전반적으로 프로세스당 전체 대역폭은 MPI 그룹에 더 많은 노드가 포함된 경우 - 모든 클러스터 노드의 MTU가 9000바이트로 증가한 경우(예: 클러스터 노드로 e-VLBI 스트리밍) - 다소 낮았다. 그럼에도 불구하고 평균적으로 호스트당 양방향 속도는 8~15Gbps였으며 InfiniBand가 없어도 여전히 8Gbps 실시간 상관관계를 지원하기에 충분함을 확인하였다.

추가적으로 OSU Micro-Benchmarks(참고: <http://mvapich.cse.ohio-state.edu/benchmarks/>)의 osu_bw(단방향) 및 osu_bw(양방향)로 노드 사이의 대역폭을 다음과 같이 테스트 하였다.

```

MPIPARAM="--report-bindings --mca btl_tcp_if_exclude lo,eml,eth0 --mca
mpi_yield_when_idle 1 --mca rmaps seq --map-by node --bind-to none"
OSU_BW=/home/difx/evlbi/sejong2020difx/osu/bin/osu_bw
OSU_BIBW=/home/difx/evlbi/sejong2020difx/osu/bin/osu_bibw
mpirun $MPIPARAM -np 2 -H belbi2,n2 $OSU_BW -m 262144:67108864
mpirun $MPIPARAM -np 2 -H n1,n2 $OSU_BW -m 262144:67108864
mpirun $MPIPARAM -np 2 -H n3,n6 $OSU_BW -m 262144:67108864
mpirun $MPIPARAM -np 2 -H belbi2,n2 $OSU_BIBW -m 262144:67108864
mpirun $MPIPARAM -np 2 -H n1,n2 $OSU_BIBW -m 262144:67108864
mpirun $MPIPARAM -np 2 -H n3,n6 $OSU_BIBW -m 262144:67108864

```

여러 가지 경우에 대하여 다양한 테스트를 네 번 실행한 결과는 다음과 같다.

Msg Size (Bytes)	Osu_bibw bidir. n1-n2 (MB/s)	Osu_bibw belbi2-n2 (MB/s)	Osu_bibw n3-n6 (MB/s)	Osu_bw single n3-n6 (MB/s)
262144	1222.46	1214.56	1221.92	1206.37

524288	1230.05	1227.33	1229.24	1221.20
1048576	1233.32	1225.17	1233.51	1230.30
2097152	1233.35	1185.71	1233.76	1232.77
4194304	1218.55	1148.71	1233.78	1180.65
8388608	1195.77	1110.18	1215.12	1137.97

이상 OSU 벤치마크에서 달성한 데이터 속도는 ~9.5Gbps였으며 최저 속도는 8.8Gbps이다.

실제로 `sysctl` 설정을 조정해도 성능이 크게 달라지지는 않았으며, `mpiBench`의 결과는 성능 측정 실행마다 차이가 크기 때문에 명확한 차별성을 확인하기 어려웠다. 반면 OSU BW 결과는 재실행을 수행하여 그 측정 결과에 보다 나은 일관성을 보였다. `sysctl` 조정은 성능에 실질적인 차이를 가져오지는 않았기 때문에, 기존 `sysctl` 설정이 적합한 것으로 판단되었다.

3. 실시간 e-VLBI 상관처리

`vdifstream`을 사용하여 VDIF 데이터를 상관기로 스트리밍하는 원격 Mark6 장치와 세종 DiFX 리눅스 클러스터(belbi2)를 이용하여 실시간 상관처리를 테스트 수행하고, belbi2에서 시작된 `statemon` 및 `errormon2`로 상관처리 진행률을 모니터링 하였다. 이 테스트는 세종 DiFX가 클러스터 과부하 또는 클러스터의 기타 병목 현상으로 인해 들어오는 원 데이터 패킷의 손실 없이 예상되는 크기의 상관처리 데이터 레코드를 생성하는지 여부를 측정하였다.

이를 위해 세종 DiFX 스토리지에서 일부 VLBI 관측 스케줄(VEX) 생성 스크립트와 DiFX(.v2d) 템플릿 파일을 생성하여 수정하고 업데이트하였다. 참고로 생성된 VEX 및 .v2d 템플릿은 별도의 데이터 스트림이 있는 1...4채널 x 512MHz 설정과 16채널 16/32MHz 단일 데이터 스트림 설정이 있다.

상관처리는 관측국 3~4기 x 1024Mbps 또는 2048Mbps, 즉 총 3Gbps~8Gbps의 클러스터 입력 속도로 테스트되었으며, 한 스캔 당 관측 시간은 10분으로 설정하고 여러 스캔을 연속으로 처리하는 것으로 구성되었다.

관측국 3기 x 1024 Mbps 테스트

이 설정은 기본적으로 1차년도에서 수행한 e-VLBI 테스트와 동일한 것으로 업그레이드 된 belbi2 DiFX 리눅스 클러스터는 데이터 손실 없이 실시간 e-VLBI 상관처리가 성공적으로 수행되었다. 참고로 belbi2의 `iptables` 기반 트래픽 포워딩은 필요하지 않다.

다양한 DiFX 프로세스 스레드(v2d 파일 `nThread=<nr>` 매개변수)에 대해 몇 가지 컴퓨팅 레이아웃을 시도했습니다. 아래와 같이 상관처리 설정을 했으며, 전반적으로 들어오는 데이터 속도는 안정적이었고, 컴퓨팅 스레드 수는 실시간 성능에 큰 차이가 없으며, 상관처리가 원활하게 수행되었다.

```
machines = belbi2, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6,
            n1,n2,n3,n4,n5,n6
nThread = 4
```

관측국 4기 x 1024 Mbps 테스트

이 설정 또한 1차년도에서 수행한 1024Mbps e-VLBI로 동일하지만, 관측국 4기를 이용하여 상관처리를 수행하였고, 업그레이드된 belbi2는 총 4096Mbps의 실시간 e-VLBI 테스트를 성공적으로 수행하였다. 위와 동일한 v2d nThreads를 사용하였으며, 데이터 손실은 발생하지 않았다.

관측국 3기 x 2048 Mbps 테스트

1차년도에서 업그레이드 되기 이전의 세종 DiFX 리눅스 클러스터 서버에서 실패하였던, 2Gbps(2048 Mbps) e-VLBI 상관처리 성능을 테스트하였다. 관측국 3기 x 2Gbps e-VLBI에서 총 네트워크 트래픽 속도는 ~6.2Gbps였으며, 관측국 3기로부터 데이터를 수신하는 DiFX 프로세스로 인해 belbi2 CPU 로드가 상당히 높게 사용되고 있음을 확인할 수 있었다. iptables 기반 패킷 포워딩 없이도 상관처리 성능은 양호하게 진행되었으며, 원 데이터의 손실도 없었다. 다음은 이 테스트에서 사용된 v2d 설정이다.

```
machines = belbi2, n1,n1,n1,n1,n1,n1,
            n2,n2,n2,n2,n2,n2,
            n3,n3,n3,n3,n3,n3,
            n4,n4,n4,n4,n4,n4
nThread = 1
nCore = 53
```

또한 .v2d DATASTREAM machine=<hostname>을 업데이트한 후 belbi2에서 클러스터 노드로 전달되는 iptables 패킷을 사용하는 것으로 설정을 업데이트 한 후 관측국 3기 x 2048Mbps 테스트도 진행하였다. 그 결과 iptables와의 "데이터 분산" 상관처리도 데이터 손실 없이 안정적으로 수행되었으며, 이 때 .v2d 파일 설정은 다음과 같다.

```
machines = belbi2, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6,
            n1,n2,n3,n4,n5,n6
nThread = 4
nCore = 53
```

관측국 4기 x 2048 Mbps 테스트

관측국 4기 x 2Gbps(2048Mbps) e-VLBI 테스트에서는 iptables 설정을 하지 않고 상관처리를 수행했을 경우, 상관처리가 제대로 수행되지 않음을 확인하였다. 총 ~8.2Gbps 데이터 입력 속도에 대해 DiFX 데이터 수신 프로세스가 belbi2에 매우 높은 부하를 야기시켰기 때문이다. 각 스캔의 처음 ~10초 정도의 데이터 상관처리는 큰 문제 없이 수행이 되었지만, 이후 데이터 손실이 증가하기 시작했고 DiFX는 때때로 관측 데이터 수신을 중단시켜서 원활한 상관처리가 수행되지 못했다.

하지만 앞에서 설명한 iptables 관련 설정을 한 이후 총 속도 8.2Gbps의 실시간 DiFX 상관처리가 각 10분 스캔 동안 안정적으로 작동하였다. 데이터 손실 또한 전혀 없었으며, 데이터 가중치는 100%였다.

이에 적용된 방화벽 규칙은 KVN 관측국에서 전송되는 VDIF 스트림을 belbi2에서 클러스터 노드 n1-n3으로 전달하였고, 세종 데이터 스트림은 belbi2에서 n4로 전달되었다. 추가로

세종 데이터 스트림을 **belbi2**를 거치지 않고 노드 **n4**로 직접 스트리밍시켜 상관처리 하는 테스트도 수행했는데, 두 경우 모두 모두 데이터 손실 없이 상관처리가 원활하게 진행됨을 확인하였다. 이 때 **DIFX** 파일의 클러스터 노드 할당 설정은 다음과 같다.

```
machines = belbi2, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6, n1,n2,n3,n4,n5,n6,  
           n1,n2,n3,n4,n5,n6  
nThread  = 4  
nCore    = 53
```

이상의 성능 측정 및 실험결과, 2차년도에서 세 개의 노드를 추가하여 성능이 업그레이드 된 세종 **belbi2** **DIFX** 리눅스 클러스터는 이전 클러스터와 달리 **4 x 2048Mbps** 실시간 **e-VLBI** 상관처리를 잘 수행할 수 있음을 확인하였다. 단, 관측국 4기 x **2Gbps e-VLBI**의 경우 **belbi2** 서버의 과부하를 피하기 위하여 앞서 기술한 **iptables** 방화벽 기반 트래픽 포워딩을 사용해야만 한다.

참고문헌

- [1] The Publications of the Astronomical Society of the Pacific, Volume 119, Issue 853, pp. 318-336.
- [2] Publications of the Astronomical Society of the Pacific, Volume 123, Issue 901, pp. 275 (2011).

부록 3. 세종 전파망원경 시스템 성능 지표

세종 VLBI 전파망원경 시스템 성능 지표

Status report 2022 of Sejong 22-m radio telescope

1. Introduction

이 문서에서는 국토정보지리원 세종 우주측지관측센터 22-m 전파망원경 (그림 1)의 K (22 GHz), Q (43 GHz) 밴드 대역 관측 성능 지표를 기술한다.

세종 우주측지관측센터 22-m 전파망원경 (hereafter 세종 망원경)은 측지 연구를 주 목적으로 세종시에 건설된 22 m 직경의 전파 안테나로 S (2 GHz), X (8 GHz), K (22 GHz), Q (43 GHz) 대역을 관측할 수 있다. 이 중 S와 X 밴드 대역은 국제 VLBI 회원국 (IVS)과의 관측 수행을 목적으로 하고 있으며, K와 Q 밴드는 한국천문연구원의 밀리미터파 전용 VLBI인 KVN (Korean VLBI Network) 및 동아시아 VLBI인 EAVN (East Asia VLBI Network)과의 합동 관측을 위해 그 전파 수신기가 제작 및 설치되었다.

따라서, K/Q band 대역 VLBI 관측을 위한 세종 전파망원경 성능에 대한 기준 지표가 필요하며, 여기에 그 성능 지표 측정 결과를 제시한다.

우선, 현재 세종 전파망원경의 좌표는 표 1과 2와 같다.

표 1. 세종 망원경의 geographic 좌표

Antenna	Latitude (° ' ")	Longitude (° ' ")	Elevation (m)
KSJ (세종망원경)	127 18 11.0	36 31 22.0	156

표 2. 세종 망원경의 geocentric 좌표

Antenna	X (m)	Y (m)	Z (m)
KSJ (세종망원경)	-3110079.9600	4082066.7340	3775076.8320

The position was measured in October 2014.



그림 1. 세종 우주측지관측센터 22-m 전파망원경

2. Antenna

세종 망원경은 Cassegrain 타입의 22-m 구경의 안테나이며, 측지 관측 연구의 목적으로 건설되었다. 주경의 구동속도는 방위각과 고도 방향 모두 $5^{\circ}/\text{sec}$ 이며, 부경은 KVN 망원경과 같은 구조로써, 주경의 중력 변형, 부경 자체의 처짐 현상들을 원격으로 보정 가능하다. 세종 망원경의 안테나 특성은 표 3과 같다.

표 3. 세종 망원경의 안테나 특성

Antenna	Parameters
Diameter	D = 22 m (Cassegrain)
Receiving frequency	S/X/K/Q (2/8/22/43 GHz)
Reflector surface accuracy	86 μm
Slewing speed	$5^{\circ}/\text{sec}$ @ Az. & El.
Max. driving acceleration	$5^{\circ}/\text{sec}^2$ @ Az. & El.
Operation range	Az.: $\pm 270^{\circ}$ deg., El: 0 – 90 deg.

3. Receiver temperature & System temperature

K/Q 밴드의 준광학계 손실을 포함하지 않은 수신기 잡음 온도를 표 4에 제시하였고, 안테나 구경 효율, 이득 곡선, 지향 정밀도 측정 관측간 Sky dipping 및 calibration 간에 얻어진 수신기 잡음 온도와 시스템 온도는 표 5와 같다.

표 4. K/Q band 수신기 잡음 온도 (T_{rx})

밴드 (GHz)	상온 흑체 (μW)	저온 흑체 (μW)	Y-factor	T_{rx} (K)
K (22)	26.17	9.05	2.89	20
Q (43)	77.60	32.00	2.43	78

표 5. 관측으로 얻어진 K/Q band 수신기 잡음 온도 (T_{rx})와 시스템 잡음 온도 (T_{sys})

밴드 (GHz)	T_{rx} (K)	T_{sys} (K)	Date
K (22)	~ 23 – 55	~ 81 – 135	2022 Jan. – Feb.
Q (43)	~ 98 – 120	~ 186 – 222	2022 Jan. – Feb.

4. Antenna sub-reflector model

전파 망원경은 관측 고도의 변화에 따라서 중력에 의한 주경 변형과 부경의 처짐 현상들이 발생할 수 있으며, 이러한 부분들은 안테나 구경 효율의 저하를 가지고 온다. 이를 보상하기 위하여 세종 망원경에는 부경 hexapod correction 기술이 활용되고 있으며, 6 개의 actuator 구성되어 있으며, 이를 통해 부경의 위치와 기울기를 제어할 수 있다. 안테나 지향 고도 변화에 따른 주경면의 중력 변형과 부경의 처짐 또는 기울임 현상은 다음과 같이 고도에 따른 삼각 함수 형태로 기술되어 질 수 있다.

$$\begin{aligned} X_{\text{ModelCor}} &= X_{\text{coeff}}[0] \\ Y_{\text{ModelCor}} &= Y_{\text{coeff}}[0] + Y_{\text{coeff}}[1] \sin(E_{\text{cmd}}) + Y_{\text{coeff}}[2] \cos(E_{\text{cmd}}) \\ &\text{(Two unknowns. } Y_{\text{coeff}}[1] \text{ is expected to be zero.)} \\ Z_{\text{ModelCor}} &= Z_{\text{coeff}}[0] + Z_{\text{coeff}}[1] \sin(E_{\text{cmd}}) + Z_{\text{coeff}}[2] \cos(E_{\text{cmd}}) \\ &\text{(Two unknowns. } Z_{\text{coeff}}[1] \text{ is expected to be zero.)} \\ \text{TiltModelCor} &= \text{Tiltcoeff}[0] \\ \text{TipModelCor} &= \text{Tipcoeff}[0] + \text{Tipcoeff}[1] \sin(E_{\text{cmd}}) + \text{Tipcoeff}[2] \cos(E_{\text{cmd}}) \\ &\text{(Two unknowns. Tipcoeff}[1] \text{ is expected to be zero.)} \end{aligned}$$

위에서 기술한 X, Y, Z, Tip, Tilt 는 각각 X, Y, Z 방향의 위치와 X 축과 Y 축을 중심으로 한 회전을 뜻하며, Xcoeff, Ycoeff, Zcoeff, Tiltcoeff, Tipcoeff 는 각 방향에 대한 삼각함수 계수를 의미하며, Ycoeff[1], Zcoeff[1], Tipcoeff[1]은 중력 방향을 고려했을 때 0 이 된다.

부경의 최적 위치를 얻기 위하여 부경의 위치를 이동시켜 가면서 신호의 강도를 측정하였다. 이로써 얻어진 고도에 따른 최적 부경 위치와 부경 제어 모델을 정리하면 다음과 같다.

$$\begin{aligned} \text{- X 방향} & & X = 1500 & & \# \text{ X 방향은 } El \text{에 대한 변화 없음.} \\ \text{- Y 방향} & & Y(El = 65) = 1200 \\ & & Y(El = 30) = -4000 \\ & & Y_{\text{ModelCor}} = 6156 - 11727 \times \cos(El) \\ \text{- Z 방향} & & Z(El = 20) = 14000 + 1350 & & \# \text{ 천체} \\ & & Z(El = 47) = 14000 & & \# \text{ W Hya} \\ & & Z(El = 70) = 14000 - 1000 & & \# \text{ V1111 Oph} \\ & & Z_{\text{ModelCor}} = 16694 - 3931 \times \cos(El) & & \# \text{ U Her} \end{aligned}$$

이 값들은 모두 2015 년에 측정되어졌고, 위 부경 모델의 식을 이용해서 고도에 따라서 부경의 위치를 자동 조절이 가능하지만, 현재 S/X 밴드 대역을 이용한 측지 관측에서는 부경 위치를 변화시키지 않고 사용하기 때문에 고도 50 도 근처로 최적화된 X, Y, Z offset 만 적용하여 사용하고 있다. Offset 값은 다음과 같다.

$$X = 1500 \mu\text{m}, Y = -1000 \mu\text{m}, Z = 14000 \mu\text{m}$$

이 경우 고도가 매우 높거나 낮은 경우 구경 효율의 감소가 예상되며 이는 이득 곡선 (gain curve)에 반영되어 나타난다.

5. Antenna beam size and Aperture efficiency

표 6은 2022년 1월, 새로 측정한 K (22 GHz LCP & RCP) 와 Q (43 GHz LCP & RCP) band의 HPBW (Half Power Beam Width), aperture efficiency, main-beam efficiency를 나타낸다. 측정 당시 Q band RCP는 시스템 문제로 측정이 불가하였다. 측정은 목성 관측을 통해서 이루어졌으며, 측정 당시 목성의 각 크기는 34', 밝기 온도는 K/Q 밴드 각각 136.2 K와 154.8 K이었다.

표 6. 세종 망원경의 빔 크기와 구경 효율

Band (GHz)	HPBW (arcsec)	Aperture eff. (%)	Beam eff. (%)	C.F. ^a (Jy/K)
K (22L)	128	68	56	11.79
K (22R)	126	71	57	11.19
Q (43L)	61	66	47	12.08
Q (43R)

* Observed date: **27 Jan. 2022**

^a indicates the conversion factor between the antenna temperature and the flux density.

Aperture efficiency는 고도에 따라 약간의 변화가 있으며, 세종 망원경의 고도에 대한 dependency는 gain curve 측정 관측을 통해서 보정되어질 수 있다. 이에 대한 내용은 다음 섹션에 제시하였다. 표 6에 제시된 값들은 gain curve coefficient가 반영이 된 망원경 최적 고도에서의 수치들을 의미한다 (K: 51 deg., Q: 61 deg.).

6. Gain Curve

중력에 의한 안테나 주경의 변형과 부경의 처짐 현상을 보상하기 위하여 부경 모델을 적용하고 있으나, 고도에 따른 효율의 저하는 일정 부분 여전히 존재하고 있다. 따라서, 신호가 강한 천체를 이용하여 안테나의 지향 고도에 따른 세기 변화를 추적하면 망원경의 표준화된 이득 곡선을 얻을 수 있다.

[그림 2](#)는 세종 망원경 K(22)/Q(43) 밴드의 이득 곡선을 보여 주고 있다. K/Q 밴드 모두 LCP 편광 모드로 설정하였으며, 각각 강한 H₂O maser 천체 (U Her) 와 SiO maser 천체 (U Ori) 를 이용하여 고도 70도에서 15도까지의 변화를 측정하였다. 현재 K/Q 밴드의 최적 고도는 각각 51 도와 61도로 측정되어졌다. [그림 3](#)은 이득 곡선 관측 간에 측정된 시스템 잡음 온도의 변화를 보여 준다.

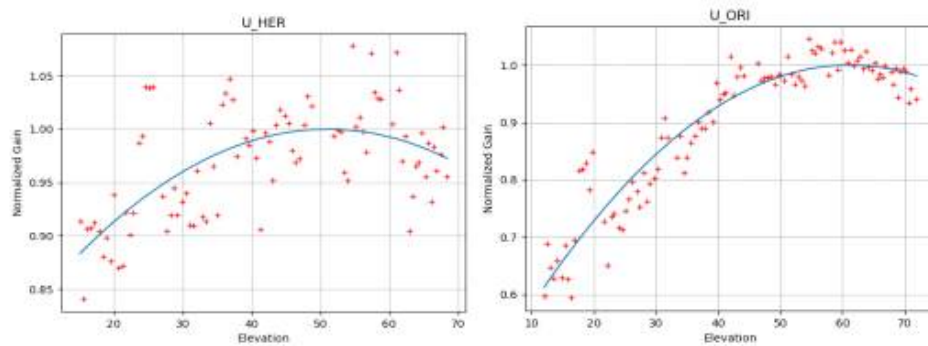


그림 2. K(22), Q(43) 밴드의 normalized gain curve. Left: H₂O maser source U Her 관측을 통해 얻어진 K(22 L) 밴드의 gain curve, Right: SiO maser source U Ori 관측을 통해 얻어진 Q(43 L) 밴드의 gain curve.

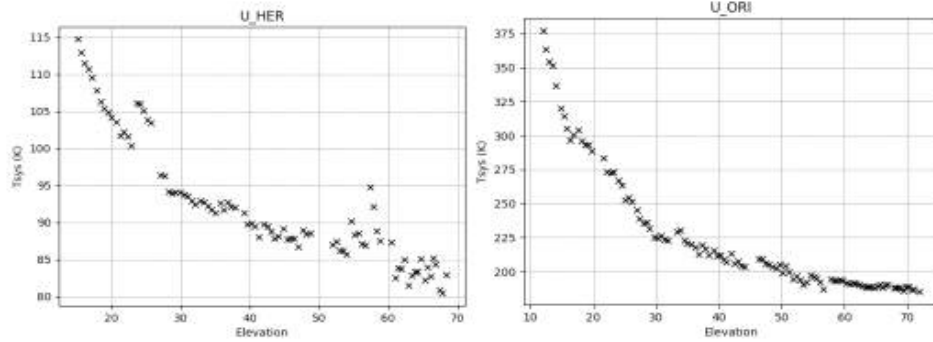


그림 3. K(22), Q(43) 밴드의 고도에 따른 system temperature (T_{sys})의 변화 추이. Left: H₂O maser source U Her 관측을 통해 얻어진 K(22 L) 밴드의 T_{sys} , Right: SiO maser source U Ori 관측을 통해 얻어진 Q(43 L) 밴드의 T_{sys} .

그림 2의 gain curve는 2차 polynomial fitting을 하였으며, 그에 대한 form은 아래와 같다 (El = Elevation).

$$G_{norm} = A0 \cdot El^2 + A1 \cdot El + A2$$

위 fitting을 통하여 얻어진 coefficient 값은 표 7과 같다.

표 7. Coefficients of normalized gain curve

Band (GHz)	A0	A1	A2	Observed Date
K (22L)	-9.02E-05	9.21E-03	7.65E-01	2022.01.18.
Q (43L)	-1.60E-04	1.96E-02	4.00E-01	2022.02.03.

6. Antenna pointing accuracy

실제 전파 망원경의 구조가 이상적이지는 않기 때문에 안테나가 지향하는 방향과 입력된 실제 방향 사이에 체계적인 차이가 있을 수 있다. 이러한 차이는 망원경 축과 실제 고도 방위각 축과의 불일치에서 생기는 오차, 굴곡이 있는 베어링 평면의 편차, 중력에 의한 안테나 경면의 편차등에 기인한다. 이로써 발생하는 지향 오차는 고도와 방위각의 함수 형태로 주어질 수 있으며 이 오차 값을 보정해 주는 것이 포인팅 모델이다. 세종 망원경의 K/Q 밴드 포인팅 모델 적성을 위하여 잘 알려져 있고 세기가 강한 여러 개의 43 GHz SiO maser 천체들을 선정하여 전 고도에 대하여 반복적 five pointing 관측을 수행하였다. 측정은 2022년 1월 10일에 수행하였으며, 그 결과

offset을 표 8에 제시하였다. 모델 적용 후, 측정한 포인팅 offset은 그림 4와 같다.

표 8. 세종 망원경의 antenna pointing offset

Total* (arcsec)	Az. (arcsec)	El. (arcsec)	Frequency (GHz)	Observed date
5.01	2.59	4.29	43.122	2022. 01.10.

* Total offset = Sqrt(Az. offset^2 + El. offset^2).

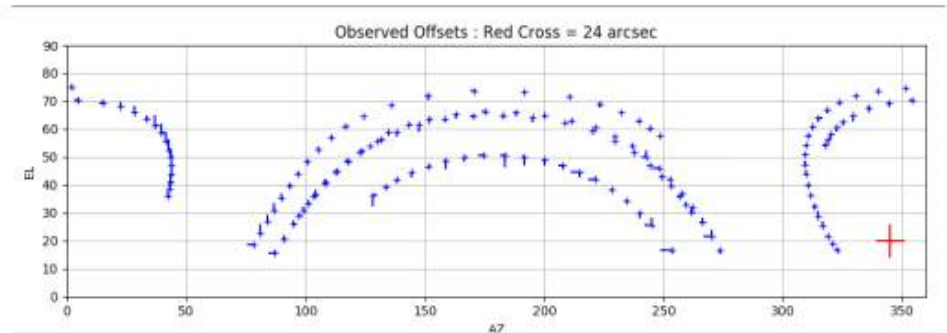


그림 4. 포인팅 모델 적용 결과

7. Receiver

세종 망원경에는 관측 주파수 대역별로 S (2 GHz), X (8 GHz), K (22 GHz), Q (43 GHz) 대역 4개의 수신기를 운영하고 있으며, 국제 저주파 측지 관측용으로 S/X 밴드, 고주파 측지/천문 관측용으로 K/Q 밴드를 사용하고 있다. 그림 5는 세종 망원경의 준광학계와 수신기 배치를 보여 주고 있다. 세종 망원경의 관측 가능 주파수 범위는 주파수를 7 - 8 GHz 대역에서 조정할 수 있는 주파수 합성기 (Frequency Synthesizer)를 도입하여 LO 주파수 source로 사용하는 경우 표 9와 같다. 이에 관한 구체적인 내용은 다음 섹션에 기술하였다.

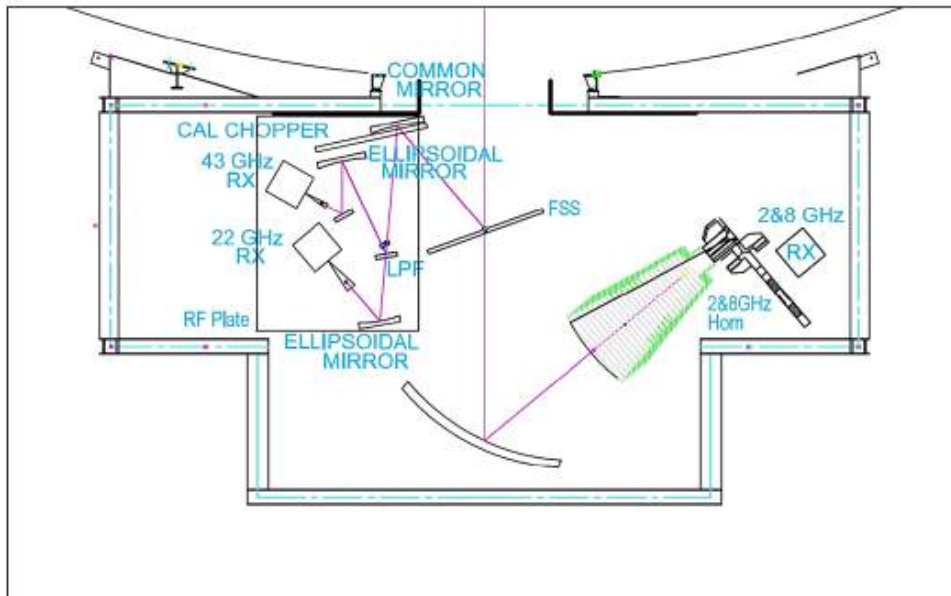


그림 5. 세종 망원경의 준광학계와 수신기 배치도

표 9. 세종 망원경의 K/Q 밴드 관측 가능 주파수 범위

Band	관측 주파수 (MHz)	LO1 (MHz)	LO2 (MHz)	net LO (MHz)	DBBC Input frequency
K	21750 – 22650	13250	8400	21650	100 – 1000
Q	42790 – 44090	34110	7380*	41500	1300 – 2600

*: 주파수 합성기 사용

8. Digital backends

세종 우주측지관측센터에서는 backend로써, 2개의 Conditioning Module (GCoMo)와 2개의 ADB3L sampler 보드, 2개의 Core3H data processing 보드로 구성되어 있는 DBBC3-2L2H (Digital Base Band Converter)를 운영하고 있다. 하나의 ADB3L 보드에는 4개의 sampler가 장착되어 있고, sampler 각각은 2048 MHz로

샘플링이 가능하다. ADB3L은 4개의 sampler에 90도 위상을 줘서 최대 8192 MHz 샘플링 자료까지 생성할 수 있다. ADB3L 출력 신호는 128 Mbit/s로 Core3H 보드로 전송된다. 그림 6은 이러한 세종 망원경 backend인 DBBC3의 구성을 보여준다.

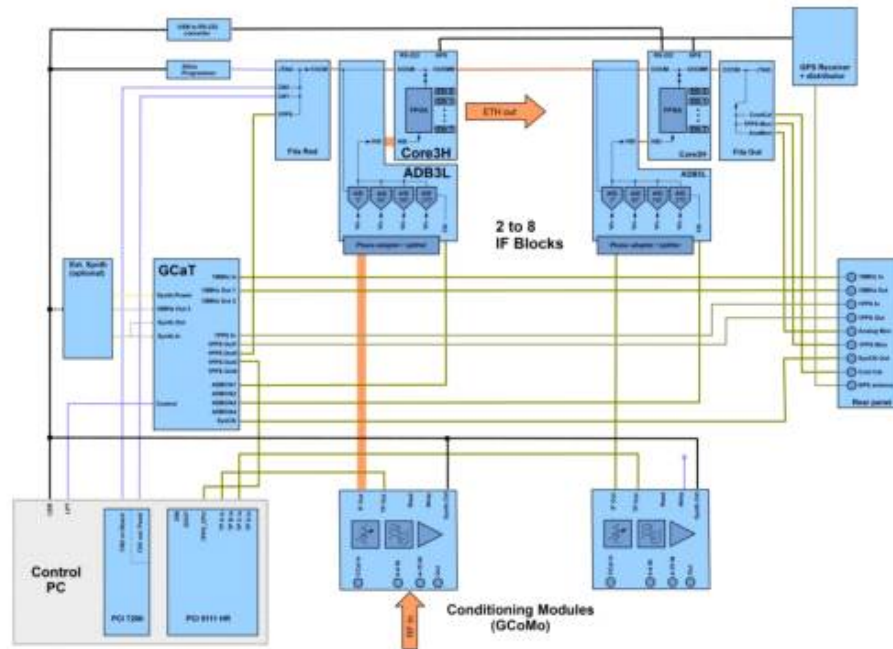


그림 6. 현 세종 망원경의 DBBC3 구성도

KVN과 같은 K/Q 밴드 동시 관측 및 유연성 있는 추가 확장을 위하여, 현재 구성되어 있는 DBBC3-2L2H (2 GCoMo+2ADB3L+2Core3H)를 DBBC3-4L4H로의 업그레이드를 추진 중에 있다. 또한, 이와 별도로 주파수 대역 분할을 통하여 현 DBBC-2L2H로 관측 가능한 IF 입력 개수를 확장하는 방법도 추진 중에 있다. 이를 위하여 입력 신호를 관측기기실로 전송하는 EMCORE 광전송장치와 DBBC3 입력단 사이에 2개의 input-combination 회로를 구성하였으며, 각 입력 신호의 세기를 측정하기 위하여 TPD (Total Power Detector) 모듈을 추가 구성하였다.

한편, 세종 망원경 Down Converter의 출력 주파수 범위는 100 - 2600 MHz 이며, DBBC3의 입력 주파수 대역(본래 0 - 4000 MHz)도 이에 한정된다. 추가 구성된

input-combination 회로의 적절한 주파수 분할은 baseband1: 100 - 1000 MHz과 baseband2: 1300 - 2600 MHz이다.

Core3H 보드는 Digital Down Conversion (DDC) 기능과 신호 양식을 변환하여 10 GbE 광케이블로 출력하는 기능을 지원한다. DDC는 16개 채널의 출력 (USB 8 + LSB 8)을 지원하며, Core3H는 4개의 10 GbE 출력에서, 현재 2 광케이블/1보드 만 연결되어 있다. [그림 7](#)은 DBBC3와 주변 장치의 연결을 보여 주고 있다.

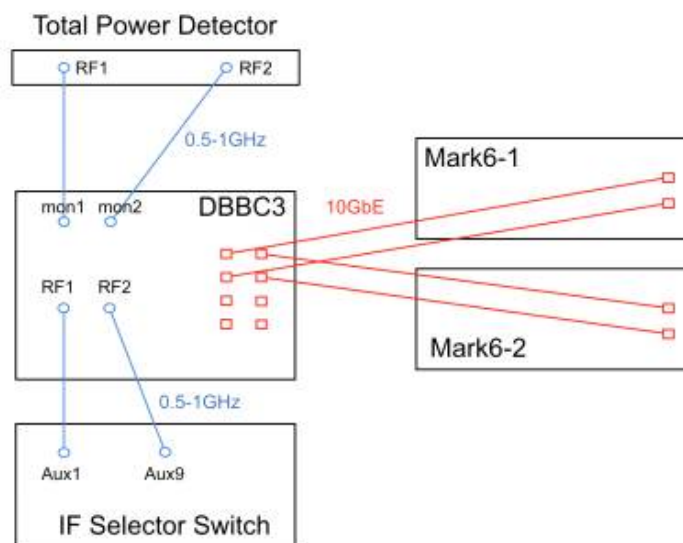


그림 7. 세종 망원경 DBBC3와 주변 장치 연결도

DBBC3는 Windows OS의 pc를 내장하고 있으며, 이 pc에서 작동하는 s/w를 통해 직접 DBBC3를 제어할 수 있다. 또한, 원격 제어를 위해 TCP/IP 통신을 통한 원격 명령어 제어 입력을 지원한다.

DBBC3 제어 컴퓨터는 2개의 네트워크 포트를 지원하며, 이 중 하나에는 LAN IP를 할당하였고, 다른 하나에는 외부에서 유지 보수 혹은 서비스를 위해 접속할 수 있도록 외부망 IP를 할당해 놓았다. 외부망의 케이블 연결은 보상을 위해 평상시는 차단되어 있으며, 필요시 한시적으로 연결할 수 있다. 관측용 컴퓨터에서는 원격 데스크탑 프로그램인 remmina를 이용해서 DBBC3 제어 컴퓨터의 화면을 관측용 컴퓨터에 불러올 수 있다 ([그림 8](#)).

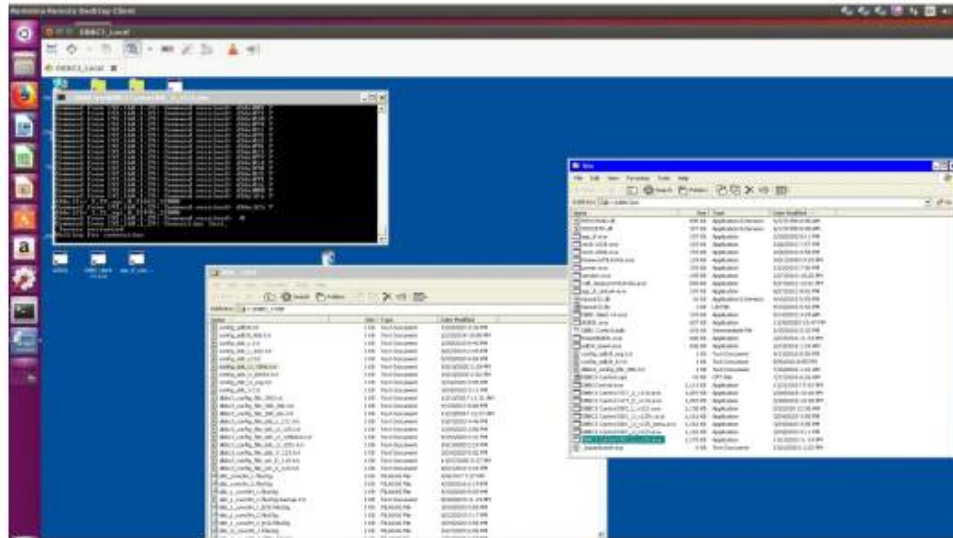


그림 8. Remmina를 통한 DBBC3 제어 화면 (DBBC3_Local)

DBBC3의 사양을 정리하면 표 10과 같다.

표 10. 세종 망원경의 DBBC3 구성 현황 및 주요 사양

구성		Specification
GCoMo	<ul style="list-style-type: none"> ● 32 dB programable attenuator ● Total power over full band ● Manual or Automatic gain control ● Direct 0 – 4 GHz input or down-converter 4 – 15 GHz 	
ADB3L	<ul style="list-style-type: none"> ● 1 × 0 – 4 GHz input ● 2048 MHz sampling clock ● 16 × 8-bit output data 	
Core3H	<ul style="list-style-type: none"> ● Digital down conversion up to 16 BBC ● Fila10G (timesync, packetizing) ● 16 bus × 8-bit ● Output rates: 2, 4, 8, or 16 Gbps ● 4 optical interface 	

DBBC3의 firmware는 DSC, DDC_L, DDC_V, DDC_U, OCT_S, OCT_D mode로 구성되어 있으며, 각 mode 별 지원 사양은 표 11과 같다. 이 중 KVGCS s/w에 포함된 DBBC3 제어 감시 s/w는 DDC_L 기능 사용에 초점이 맞춰져 있다. 다른 OCT모드에 대해서는 추가 개발이 필요한 상황이다. DBBC3를 관측에 활용하기 위해서는 DBBC3를 구성하는 각 보드의 기능을 제어하고 상태를 확인하는 기능이 구현되어야 한다. DBBC3 s/w는 최상위 DBBC 제어 소프트웨어를 통해 Core3H, ADB3L, Synthesizer 등 내장 보드를 제어하도록 되어있으며, 각 보드를 제어하는 명령어와 명령어 양식이 따로 정의가 되어 있고 명령어 체계가 통일되어 있지 않기 때문에 각각 개별 인터페이스 모듈을 개발해야 한다. 이로 인하여 KVGCS에서의 인터페이스 모듈도 DBBC3 원격 제어를 위해 DBBC3.py, ADB3L.py, Core3H.py로 모듈화 하였다.

표 11. DBBC3 firmware 별 지원 기능

구성	Specification
DSC	<ul style="list-style-type: none"> ● Direct sampling conversion ● Full 4 GHz bandwidth/IF
DDC_L (Legacy)	<ul style="list-style-type: none"> ● Digital tunable down-conversion ● 16 BBCs/IF with fully tunable frequency ● BW: 2, 4, 8, 16, 32 MHz
DDC_V (VGOS)	<ul style="list-style-type: none"> ● 32 MHz filters ● 8 BBCs/IF
DDC_U (Unified)	<ul style="list-style-type: none"> ● BW: 1, 2, 4, 8, 16, 32, 128 MHz ● Up to 16 BBCs/IF ● Under development
OCT_S	<ul style="list-style-type: none"> ● Single 32-tap FIR-filter/IF ● 512 MHz BW: 0 - 512, 512 - 1024, ..., 3584 - 4096 ● 1024 MHz BW: 0 - 1024, ..., 3072 - 4096 ● 2048 MHz BW: 0 - 2048, 2048 - 4096
OCT_D	<ul style="list-style-type: none"> ● Double 32-tap FIR-filter/IF ● 512 MHz BW: 0 - 512, 512 - 1024, ..., 3584 - 4096 ● 1024 MHz BW: 0 - 1024, ..., 3072 - 4096 ● 2048 MHz BW: 0 - 2048, 2048 - 4096

부록 4. 주파수 합성기 프로그램

```
#include <Wire.h>
#include <Adafruit_ADS1015.h>

//ADDR 핀에 (아무것도 연결되어 있지 않을 때 ??) 기본 i2c 주소(0x48)를 사용하여 동작합니다.
Adafruit_ADS1015 ads1015;

//ADDR 핀을 이용하여 i2c 주소를 변경하였을 때에는 아래 소스코드를 이용합니다.
//Adafruit_ADS1015 ads1015(0x49);

const int clockPin = 8;
const int dataOutPin = 9;
const int selectPin = 10;
const int Div = 2;
const int Byp = 3;
const int N2 = 4;
const int N1 = 5;
const int N0 = 6;
const int S0 = 7;
const int E8 = 8;
const int E9 = 9;
const int E10 = 10;
const int INV = 11;
const int LD = 12;
const int AMP_SEL = 13;
const int Lock = 0;
//Pin 14 = Serial Logic Ground

// A0, A1, A2, A3 아날로그 포트 추가 =====
const int FM_ADC = A0;
const int RF_ADC = A1;
const int IF_ADC = A2;
const int REF_ADC = A3;
const int LV_ADC = A4;
// const unsigned int freqcode=0;
//=====

//char sendArray[6]; // decimal digit for 16 bit number (max:65535)
char freq_100MHz[8];
char buf[8];
int counter;
int b=0;
int i=0;
int x=0;
unsigned int freqcode=0;
int DivRatio=0;
int Inver=0;

void setup() {
  // put your setup code here, to run once:
  pinMode(clockPin, OUTPUT);
  pinMode(dataOutPin, OUTPUT);
  pinMode(selectPin, OUTPUT);
  pinMode(Div, OUTPUT);
  pinMode(Byp, OUTPUT);
  pinMode(N2, OUTPUT);
  pinMode(N1, OUTPUT);
  pinMode(N0, OUTPUT);
  pinMode(S0, OUTPUT);
  pinMode(INV, OUTPUT);
  pinMode(LD, INPUT);
```

```

pinMode(AMP_SEL, OUTPUT);

    // A0, A1, A2, A3 아날로그 포트 방향 설정 =====
    //pinMode(FM_ADC, INPUT);
    //pinMode(RF_ADC, INPUT);
    //pinMode(IF_ADC, INPUT);
    //pinMode(REF_ADC, INPUT);
//pinMode(LV_ADC, INPUT);
//=====

digitalWrite(selectPin, HIGH); //init selectPin high
Serial.begin(38400);
ads1015.begin();

int Freq_init=13200;
// unsigned int freqcode=(Freq_init-12000.)/0.091554;
// freqcode=(Freq_init-12000.)/0.091554;
freqcode=(Freq_init-10000.)/0.091554/10.*6.;

sendFreq(freqcode); // send frequency to YIG driver
delay(10);

findDivRatio(Freq_init); // find frequency division ratio in MHz
delay(10);

writeDivRatio(DivRatio, Inver); // set frequency division ratio to HMC705
delay(200);
// delay(40);

readADC(freqcode, DivRatio, Inver);
delay(10);
// Serial.println(" 12000 MHz initial set");

//Gain을 조정하여 최대 6.144V의 전압에 대한 데이터를 출력하며, 1 bit당 3mV의 전압으로 인식
합니다.
// ads1015.setGain(GAIN_TWOTHIRDS);
ads1015.setGain(GAIN_ONE);
// Serial.println("ADC Range: +/- 6.144V (1 bit = 3mV)");
// Serial.println(" ADC Range: +/- 4.096V (1 bit = 2mV)");

//
// ads.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV
// ads.setGain(GAIN_ONE); // 1x gain +/- 4.096V 1 bit = 2mV
// ads.setGain(GAIN_TWO); // 2x gain +/- 2.048V 1 bit = 1mV
// ads.setGain(GAIN_FOUR); // 4x gain +/- 1.024V 1 bit = 0.5mV
// ads.setGain(GAIN_EIGHT); // 8x gain +/- 0.512V 1 bit = 0.25mV
// ads.setGain(GAIN_SIXTEEN); // 16x gain +/- 0.256V 1 bit = 0.125mV

while(!Serial);
//Serial.println("Enter a frequency offset from base frequency");
}
void loop() {
    // put your main code here, to run repeatedly:
    while (Serial.available() > 0) {
        //read all char's to send from Serial number
        delay(20); //wait for Serial port
        freq_100MHz[i] = Serial.read();
        int aa = atoi(freq_100MHz) * 100;

        i = i + 1;
        delay(1); //wait for Serial port
        //when all char's received, call Serial send frequency routine
        if (Serial.available() == 0 && aa != 30000) {

            freqcode = (aa - 10000.) / 0.15259;
            //freqcode is frequency expressed by 16 bit number for 10000-20000 MHz.

```

```

    sendFreq(freqcode);// send frequency to YIG driver
    delay(10);

    findDivRatio(aa);// find frequency division ratio
    delay(10);

    writeDivRatio(DivRatio,Inver);// set frequency division ratio to HMC705
    delay(200);

    readADC(freqcode,DivRatio,Inver);
    delay(10);
}
//=====
else if (Serial.available()==0 && aa==30000){
    readADC(freqcode,DivRatio,Inver);
    delay(10);
}
}
i=0;
}

// find Division ratio and inversion
void findDivRatio(int aa)
{
    if(aa==11300||aa==12700||aa==14100||aa==15500||aa==16900||aa==18300||aa==19700)
    {
        DivRatio=1;
        Inver=0;
    }
    else if(aa==10000||aa==11400||aa==12800||aa==14200||aa==15600||aa==17000||aa==18400||aa==19800)
    {
        DivRatio=2;
        Inver=0;
    }
    else if(aa==10100||aa==11500||aa==12900||aa==14300||aa==15700||aa==17100||aa==18500||aa==19900)
    {
        DivRatio=3;
        Inver=0;
    }
    else if(aa==10200||aa==11600||aa==13000||aa==14400||aa==15800||aa==17200||aa==18600||aa==20000)
    {
        DivRatio=4;
        Inver=0;
    }
    else if(aa==10300||aa==11700||aa==13100||aa==14500||aa==15900||aa==17300||aa==18700)
    {
        DivRatio=5;
        Inver=0;
    }
    else if(aa==10400||aa==11800||aa==13200||aa==14600||aa==16000||aa==17400||aa==18800)
    {
        DivRatio=6;
        Inver=0;
    }
    else if(aa==11100||aa==12500||aa==13900||aa==15300||aa==16700||aa==18100||aa==19500)
    {
        DivRatio=1;
        Inver=1;
    }
    else if(aa==11000||aa==12400||aa==13800||aa==15200||aa==16600||aa==18000||aa==19400)
    {
        DivRatio=2;
        Inver=1;
    }
    else if(aa==10900||aa==12300||aa==13700||aa==15100||aa==16500||aa==17900||aa==19300)

```



```

{
    DivRatio=3;
    Inver=1;
}
else if(aa==10800|aa==12200|aa==13600|aa==15000|aa==16400|aa==17800|aa==19200)
{
    DivRatio=4;
    Inver=1;
}
else if(aa==10700|aa==12100|aa==13500|aa==14900|aa==16300|aa==17700|aa==19100)
{
    DivRatio=5;
    Inver=1;
}
else if(aa==10600|aa==12000|aa==13400|aa==14800|aa==16200|aa==17600|aa==19000)
{
    DivRatio=6;
    Inver=1;
}
}

// Frequency division ratio 설정 =====
void writeDivRatio(int n, int m)
{
    //sprintf(buf,"%d",n);
    //Serial.print(buf);
    // Serial.println(n);
    //sprintf(buf,"%d",m);
    //Serial.print(buf);
    // Serial.println(m);
    if(n==1)
    {
        digitalWrite(Div, LOW);//Div is High when division ratio is 2 or 3
        delay(1);
        digitalWrite(Byp, HIGH);// Byp is High only when dividson ratio is 1.
        delay(1);
        digitalWrite(N2, LOW);// N2 is always Low.
        delay(1);
        digitalWrite(N1, LOW);// N1 is High only when division ratio is 6.
        delay(1);
        digitalWrite(N0, LOW);// N0 is High only when division ratio is 4 or 5.
        delay(1);
        digitalWrite(S0, LOW);// N2 is High only when division ratio is 3 or 5.
        delay(1);
        digitalWrite(AMP_SEL, LOW);// AMP_SEL is Low when division ratio is 1-3.
        delay(1);
        if(m==0)
        {
            digitalWrite(INV, LOW);
            delay(1);
        }
        else
        {
            digitalWrite(INV, HIGH);
            delay(1);
        }
    }
    else if(n==2)
    {
        digitalWrite(Div, HIGH);//Div is High when division ratio is 2 or 3
        delay(1);
        digitalWrite(Byp, LOW);// Byp is High only when dividson ratio is 1.
        delay(1);
        digitalWrite(N2, LOW);// N2 is always Low.
        delay(1);
    }
}

```

```

digitalWrite(N1, LOW); // N1 is High only when division ratio is 6.
delay(1);
digitalWrite(N0, LOW); // N0 is High only when division ratio is 4 or 5.
delay(1);
digitalWrite(S0, LOW); // N2 is High only when division ratio is 3 or 5.
delay(1);
digitalWrite(AMP_SEL, LOW); // AMP_SEL is Low when division ratio is 1-3.
delay(1);
if(m==0)
{
    digitalWrite(INV, LOW);
    delay(1);
}
else
{
    digitalWrite(INV, HIGH);
    delay(1);
}
}
else if(n==3)
{
    digitalWrite(Div, HIGH); // Div is High when division ratio is 2 or 3
    delay(1);
    digitalWrite(Byp, LOW); // Byp is High only when division ratio is 1.
    delay(1);
    digitalWrite(N2, LOW); // N2 is always Low.
    delay(1);
    digitalWrite(N1, LOW); // N1 is High only when division ratio is 6.
    delay(1);
    digitalWrite(N0, LOW); // N0 is High only when division ratio is 4 or 5.
    delay(1);
    digitalWrite(S0, HIGH); // N2 is High only when division ratio is 3 or 5.
    delay(1);
    digitalWrite(AMP_SEL, LOW); // AMP_SEL is Low when division ratio is 1-3.
    delay(1);
    if(m==0)
    {
        digitalWrite(INV, LOW);
        delay(1);
    }
    else
    {
        digitalWrite(INV, HIGH);
        delay(1);
    }
}
else if(n==4)
{
    digitalWrite(Div, LOW); // Div is High when division ratio is 2 or 3
    delay(1);
    digitalWrite(Byp, LOW); // Byp is High only when division ratio is 1.
    delay(1);
    digitalWrite(N2, LOW); // N2 is always Low.
    delay(1);
    digitalWrite(N1, LOW); // N1 is High only when division ratio is 6.
    delay(1);
    digitalWrite(N0, HIGH); // N0 is High only when division ratio is 4 or 5.
    delay(1);
    digitalWrite(S0, LOW); // N2 is High only when division ratio is 3 or 5.
    delay(1);
    digitalWrite(AMP_SEL, HIGH); // AMP_SEL is High when division ratio is 4-6.
    delay(1);
    if(m==0)
    {
        digitalWrite(INV, LOW);
        delay(1);
    }
}

```

```

    }
    else
    {
        digitalWrite(INV, HIGH);
        delay(1);
    }
}
else if(n==5)
{
    digitalWrite(Div, LOW); // Div is High when division ratio is 2 or 3
    delay(1);
    digitalWrite(Byp, LOW); // Byp is High only when division ratio is 1.
    delay(1);
    digitalWrite(N2, LOW); // N2 is always Low.
    delay(1);
    digitalWrite(N1, LOW); // N1 is High only when division ratio is 6.
    delay(1);
    digitalWrite(N0, HIGH); // N0 is High only when division ratio is 4 or 5.
    delay(1);
    digitalWrite(S0, HIGH); // N2 is High only when division ratio is 3 or 5.
    delay(1);
    digitalWrite(AMP_SEL, HIGH); // AMP_SEL is High when division ratio is 4-6.
    delay(1);
    if(m==0)
    {
        digitalWrite(INV, LOW);
        delay(1);
    }
    else
    {
        digitalWrite(INV, HIGH);
        delay(1);
    }
}
else if(n==6)
{
    digitalWrite(Div, LOW); // Div is High when division ratio is 2 or 3
    delay(1);
    digitalWrite(Byp, LOW); // Byp is High only when division ratio is 1.
    delay(1);
    digitalWrite(N2, LOW); // N2 is always Low.
    delay(1);
    digitalWrite(N1, HIGH); // N1 is High only when division ratio is 6.
    delay(1);
    digitalWrite(N0, LOW); // N0 is High only when division ratio is 4 or 5.
    delay(1);
    digitalWrite(S0, LOW); // N2 is High only when division ratio is 3 or 5.
    delay(1);
    digitalWrite(AMP_SEL, HIGH); // AMP_SEL is High when division ratio is 4-6.
    delay(1);
    if(m==0)
    {
        digitalWrite(INV, LOW);
        delay(1);
    }
    else
    {
        digitalWrite(INV, HIGH);
        delay(1);
    }
}
}

//      A0, A1, A2, A3 아날로그 포트 읽기 함수 =====
void readADC(unsigned int freqcode, int n, int m)
{

```

```

// put your main code here, to run repeatedly:
float adc0, adc1, adc2, adc3;

// 각 채널의 입력 신호를 읽고 변수에 저장합니다.
delay(1);
adc0= ads1015.readADC_SingleEnded(0);
delay(1);
adc1= ads1015.readADC_SingleEnded(1);
delay(1);
adc2= ads1015.readADC_SingleEnded(2);
delay(1);
adc3= ads1015.readADC_SingleEnded(3);
delay(1);

// Serial.print("AIN0: "); Serial.println(adc0);
// Serial.print("FM Tuning Voltage: "); Serial.println(adc0*2/1000);
String str0=String(adc0*2/1000);
// Serial.print("AIN1: "); Serial.println(adc1);
// Serial.print("YIG IF Voltage: "); Serial.println(adc1*2/1000);
String str1=String(adc1*2/1000);
// Serial.print("AIN2: "); Serial.println(adc2);
// Serial.print("Ref Voltage: "); Serial.println(adc2*2/1000);
String str2=String(adc2*2/1000);
// Serial.print("AIN3: "); Serial.println(adc3);
// Serial.print("YIG RF Voltage: "); Serial.println(adc3*2/1000);
String str3=String(adc3*2/1000);
int Lock = digitalRead(LD);
String strLD=String(Lock);
String strFreq=String((freqcode*0.091554*10./6.+10000.));
String strFreqCode=String(freqcode);
String strSum = strFreq+' '+strFreqCode+' '+str0+' '+str1+' '+str2+' '+str3+' '+strLD;
Serial.println(strSum);

delay(10);
}

//      int _fmADC = 0;
//      _fmADC = analogRead(FM_ADC);
//      float realbuf = float(_fmADC)*5./1024.;
//Serial.println(float(_fmADC)*5./1024.);
//Serial.println(_fmADC);
//=====

// 이 부분부터는 외부기기로 설정된 주파수를 보내는 함수
//send / receive command routine
void sendFreq(unsigned int freqcode)
{
    unsigned int bb=0;

    digitalWrite(selectPin,LOW); //selectPin Low and hold low for as long as you are
transmitting
    delay(1);
    digitalWrite(dataOutPin, LOW); // set data out low when done
    delay(1);
    digitalWrite(clockPin, LOW); //set clock low when done
    delay(1);

    // original sentence int bb=atoi(sendArray);
    if(freqcode < 65436){
        bb= freqcode+0;
        //bb= freqcode;
    }
    else {
        bb= freqcode;
    }
    for (counter=0;counter!=16;counter++)

```

```

{
    delay(10);
    int b=!!(bb & (1<<(15-counter)));
    if (b==1)
    {
        digitalWrite(dataOutPin,HIGH);
        //Serial.print("sendArray[i]=");
        //Serial.println(sendArray[counter]);
        delay(20);
    }
    else
    {
        digitalWrite(dataOutPin,LOW);
        //Serial.println("sendArray[i]=");
        //Serial.println(sendArray[counter]);
        delay(20);
    }
    //toggle clock line
    digitalWrite(clockPin,HIGH);
    delay(10);
    digitalWrite(clockPin,LOW);
    delay(10);
}
delay(5); //delay between chars for byte visibility on Dig. scope
digitalWrite(selectPin, HIGH); // pull the selectPin to save the data
digitalWrite(dataOutPin, LOW); // set data out low when done
digitalWrite(clockPin, LOW); //set clock low when done
i=0;
delay(10); //write for a x uS etween write/read
}

void printComm(unsigned int freqcode)
{
    // Serial.print("Command Sent=");
    //for (x=0;x<5;x++)
    for (x=0;x<1;x++)
    {
        //Serial.print(sendArray[x]);
        //Serial.print(freqcode);
        //delay(1);
    }
    //Serial.println();
    //Serial.println("End ");
    //Serial.println();
    delay(1);
}
}

```

부록 5. 주파수 합성기 점검 매뉴얼

1) 전원 연결 및 전원 켜기와 끄기

전원을 켜기 전에 먼저, 전원 케이블이 연결되어 있어야 한다. 다음 그림 A-1의 주파수 합성기 정면 사진의 왼쪽에 위치한 Power Input 커넥터와 그림 A-2의 DC Power Supply 사이에 전원 케이블을 연결하면 된다. 전원 케이블은 +15Vdc, -15Vdc, +5Vdc 공급을 위한 각 2개의 선과 케이블 shield 접지 선을 합쳐 총 7개의 선으로 구성되어 있다. 주파수 합성기의 +15Vdc의 소모 전류는 설정 주파수에 따라 1.5~2 A 정도이다. 그림 A-2에는 주파수 합성기 이외에 1.4GHz PLO 공급 전류와 협쳐서 나타난 값이다.

전원을 켜는 순서는 그림 A-2, DC Power Supply 의 Main Power Switch를 켜고, Output On/Off Switch를 켜고, 다음에 그림 A-1의 주파수 합성기 정면에 있는 Power Switch를 켜면 된다. 전원 공급 상태 확인은 주파수 합성기 Power LED에 빛이 나오는 것으로 확인할 수 있다. 주파수 합성기 Lock 상태가 정상인지는 Lock LED를 확인하면 된다.

전원을 끌 때는 역순으로 주파수 합성기 Power Switch를 끄고, DC 전원 공급기의 Output On/Off Switch, Main Power Switch를 끄면 된다.



그림 A-1. 주파수 합성기 정면 사진; 왼쪽에 있는 Power Switch를 올리면 전원을 켤 수 있다. 정상 작동 시에 Power LED와 Lock LED에서 빛이 나옴.



그림 A-2. 주파수 합성기용 DC Power Supply 정면 사진; 수신기가 설치된 Receiver Plate의 아래에 뒤집혀서 설치되어 있다. 왼쪽에 +15V, 가운데에 -15V, 오른쪽에 +5V를 연결한다. +15V와 -15V는 2개의 Voltage 조정 다이얼을 돌려서 전압을 조정할 수 있다. Current limit는 Current 다이얼을 돌려서 조정할 수 있다. +15V는 주파수 합성기 설정 주파수에 따라 다르지만, 소모전류가 1.5~2A 범위이며, -15V 소모전류는 1A 미만이며, 5V 소모전류는 별도로 표시되지 않는다. 사진에 표시된 +15V 소모전류는 1.4GHz PLO 소모전류가 포함된 값이다.

2) 기준신호 입력 연결 및 전력 확인

그림 A-3의 주파수 합성기 후면에 100MHz와 1400MHz 입력 포트가 있다. 입력 기준 신호의 전력은 $-5\text{ dBm} \sim +5\text{ dBm}$ 이면 된다. 입력 신호의 크기와 주파수를 확인한 후에 신호선을 연결하면 된다. 입력 기준신호가 흔들리거나 기준신호 주파수 근처에 다른 주파수 신호가 있으면 주파수 합성기가 정상 동작을 하지 않는다. 기준신호 주파수 근처에 다른 주파수 신호가 있는지 보기 위해서 스펙트럼 분석기의 SPAN을 수 백MHz 이상 넓게 측정을 해서 확인해야 한다. 또한 주파수가 안정되어 있는지를 확인하기 위해서는 SPAN을 수 백 Hz \sim 수 kHz 범위로 설정하고 100MHz 신호와 1400MHz 신호가 흔들리지 않는지 확인해야 한다. 정밀한 주파수 측정을 위해서는 측정에 사용되는 스펙트럼 분석기의 기준신호를 Ref Signal Distributor의 10MHz 신호와 연결하고, 스펙트럼 분석기를 EXT Ref로 설정하는 것이 좋다.

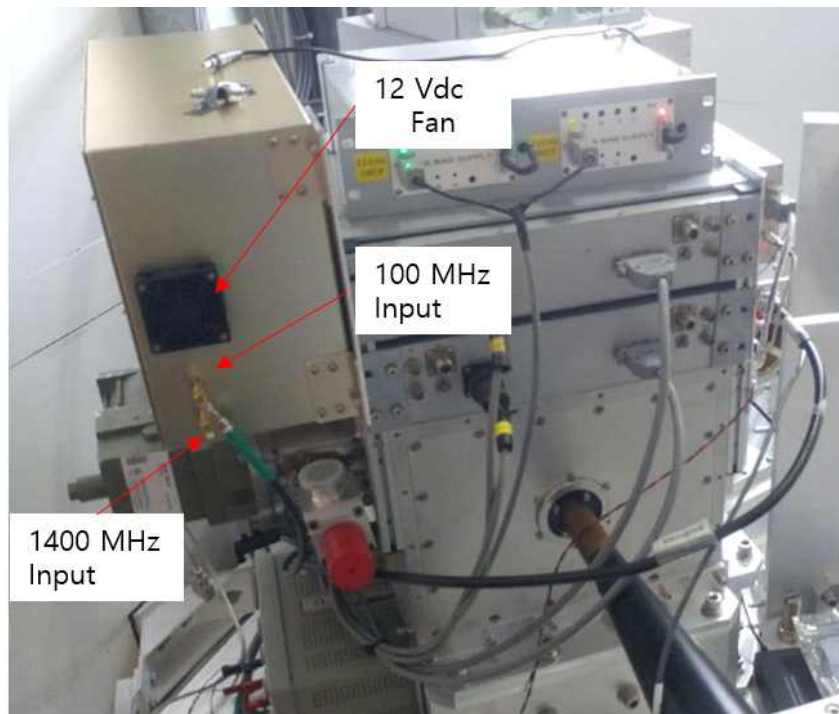


그림 A-3. 주파수 합성기 후면의 100MHz, 1400MHz 기준 신호 입력; 두 신호 모두 $-5\text{ dBm} \sim +5\text{ dBm}$ 범위의 전력이 공급되어야 한다.

3) 출력 주파수 설정과 상태 모니터

주파수 합성기에서 직접 주파수를 조정할 수 있다. 그림 A-1의 오른쪽에 있는 Control용 Dsub-9핀 커넥터에 RS232 to USB 케이블을 통해 PC에 연결한 후, RS232 통신을 통해 100MHz 단위의 세자리 숫자로 된 주파수 명령을 보내면 된다. 주파수 명령은 표 A-1의 설정 주파수를 소수점 없이 보내면 된다, 즉, 10.1GHz로 설정을 원하면, 101, 12GHz 설정을 원하면 120으로 보내면 된다. 주파수 명령을 보내면 표 A-2와 같이 상태 값을 받을 수 있다. 표 A-2에는 RS232 통신을 통해 주파수 합성기에 13.7GHz 명령인 137을 보냈을 때의 응답이다. 응답은 한줄로 되어 있으며, 7개의 숫자로 구성되어 있다. FM tuning 전압이 정상 범위인지, Lock이 되었는지 확인할 수 있다. 주파수 합성기를 켜올 때 Default 주파수인 13.2GHz로 설정된다.

표 A-1. 10-20GHz YIG 주파수 합성기의 출력 주파수 목록

1.4GHz harmonic number(M)	IF Frequency($\pm 0.1 \times N$) [GHz]											
	-0.6	-0.5	-0.4	-0.3	-0.2	-0.1	0.1	0.2	0.3	0.4	0.5	0.6
	Synthesizer Output Frequency [GHz]											
7								10	10.1	10.2	10.3	10.4
8	10.6	10.7	10.8	10.9	11	11.1	11.3	11.4	11.5	11.6	11.7	11.8
9	12	12.1	12.2	12.3	12.4	12.5	12.7	12.8	12.9	13	13.1	13.2
10	13.4	13.5	13.6	13.7	13.8	13.9	14.1	14.2	14.3	14.4	14.5	14.6
11	14.8	14.9	15	15.1	15.2	15.3	15.5	15.6	15.7	15.8	15.9	16
12	16.2	16.3	16.4	16.5	16.6	16.7	16.9	17	17.1	17.2	17.3	17.4
13	17.6	17.7	17.8	17.9	18	18.1	18.3	18.4	18.5	18.6	18.7	18.8
14	19	19.1	19.2	19.3	19.4	19.5	19.7	19.8	19.9	20		

표 A-2. 주파수 합성기 응답 명령; RS232 통신을 통해 13.7GHz 주파수 설정을 했을 때 주파수 합성기에서 PC로 보내는 신호 및 정상 범위

명령 예	명령 설명	정상 범위
13699.85	주파수 근사값, 실수	
24247	16 진수 주파수 코드 (0:10GHz, 65535:20GHz)	0-65535
2.55	FM tuning 전압	1.5-3.5
3.69	IF 신호 크기	-5dBm: 1.85, -15dBm: 1.45
0.25	100MHz 기준 신호 크기	-5dBm: 1.85, -15dBm: 1.45
0.87	YIG 출력 신호의 10 dB coupled RF 신호 크기	0.2-1
1	Lock 여부	1: 정상, 0: unlock

4) 출력 전력 확인 및 연결

그림 A-1의 정면에 있는 RF Out 포트에서 출력 전력을 확인할 수 있다. Power Meter/Sensor를 사용하여 주파수 합성기의 출력 전력을 측정할 수 있다. Power Meter/Sensor의 Calibration을 실행한 후, Power Sensor를 RF Out 포트에 연결하여 전력을 측정하면 된다. 다음 그림 A-4에 주파수 합성기 출력 전력 값을 표시하였다. 설정 주파수를 RS232 통신을 통해 바꾸면서, 표시된 값과 측정한 값의 차이가 1-2 dB 이내로 유사하게 나타나는지를 살펴보면 된다.

주파수 측정은 스펙트럼 분석기를 사용한다. 주파수 합성기와 스펙트럼 분석기를 RF 케이블로 연결한 후, 설정 주파수를 바꾸면서 스펙트럼 분석기 화면에서 설정 주파수와 같은 신호가 보이는지 확인하면 된다. 스펙트럼 분석기의 SPAN을 수 kHz 범위로 설정하여 주파수가 흔들리지 않는지 확인해야 한다. 스펙트럼 분석기에 Ref Signal Distributor의 10MHz와 연결하고, Ext Ref로 설정한 상태로 측정해야 정확한 주파수를 측정할 수 있다. 스펙트럼 분석기로도 전력을 측정할 수 있다. 케이블 손실을 고려하여 그림 A-4의 출력 전력값과 2 dB 이내로 유사한 값이 측정되는지 확인한다.



그림 A-4 10-20GHz YIG 주파수 합성기의 출력 전력 값

5) 주파수 합성기 Fan 및 내부 점검

주파수 합성기 정면과 후면에는 그림 A-1, A-3에서와 같이 12V DC FAN 1개씩이 각각 설치되어 있다. DC FAN(Orion 사의 OD6025-12HB)은 주파수 합성기 내부의 regulator board의 -12V와 단자대를 거쳐서 연결되어 있다. DC FAN이 작동을 멈추었을 때는 주파수 합성기의 뚜껑을 열어서 전원선을 분리한 후 새 DC Fan을 설치하면 된다. DC Fan은 주파수 합성기 외부에 설치되어 있고, 전원선은 주파수 합성기의 지름 2mm 구멍을 통해서 내부의 단자대와 연결해야 한다.

주파수 합성기 내부의 사진을 그림 A-5에 보였다. 외부 점검에 이상이 있을 때나 유지보수 기간에 내부 단자와 RF 커넥터 연결 상태를 확인할 필요가 있다. Regulator 보드의 입력 $\pm 15V$ 전압과 출력 $\pm 12V$, $\pm 5V$ 전압을 측정할 수 있다. RF 상자에는 5V DC Fan이 설치되어 있다. 이 FAN의 동작 여부도 확인한다. RF 커넥터가 조임 상태가 안 좋으면, 주파수 합성기 출력 전력이 불안정해 진다. 그림 A-6의 RF 상자 내부도 유지보수 기간에 내부 RF 커넥터 연결 상태를 확인하면 된다.



그림 A-5. 주파수 합성기 내부 점검 항목; Regulator 보드 입, 출력 전압, 내부 단자들과 RF 부품들의 고정 상태와 커넥터의 연결 상태를 확인한 후, 풀린 부분이 있으면 조인다

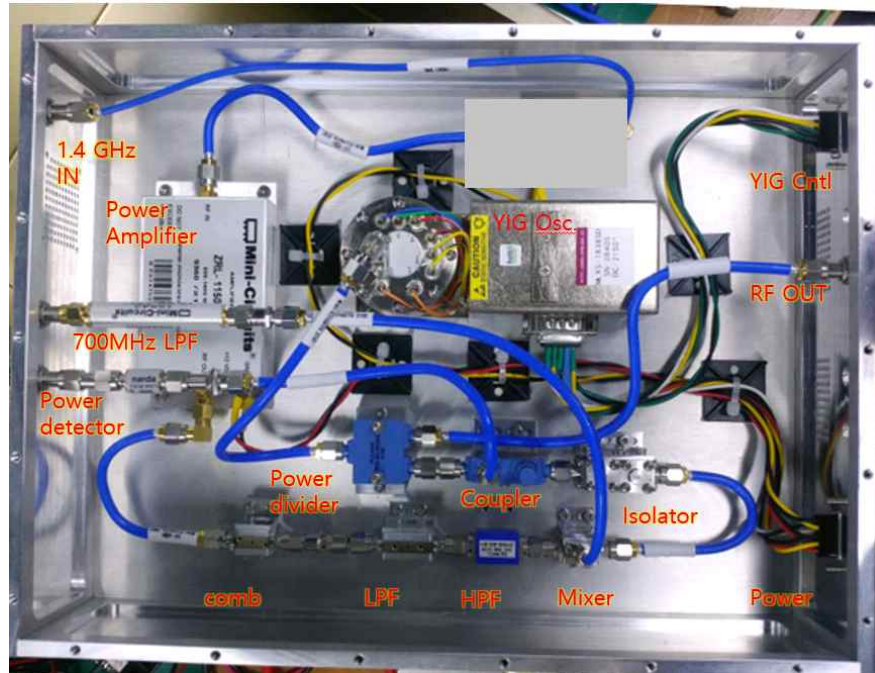
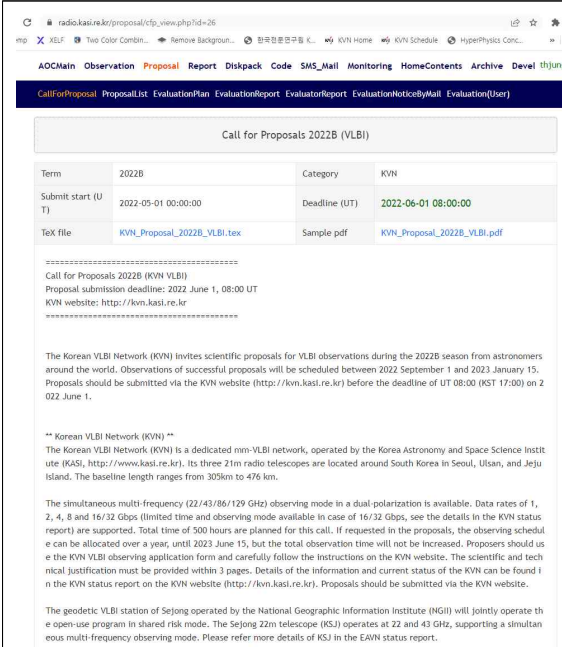


그림 A-5. RF 상자 내부의 부품들; 내부 부품들의 조립 상태와 커넥터의 조임 상태를 확인하고, 풀린 부분이 있으면 조인다.

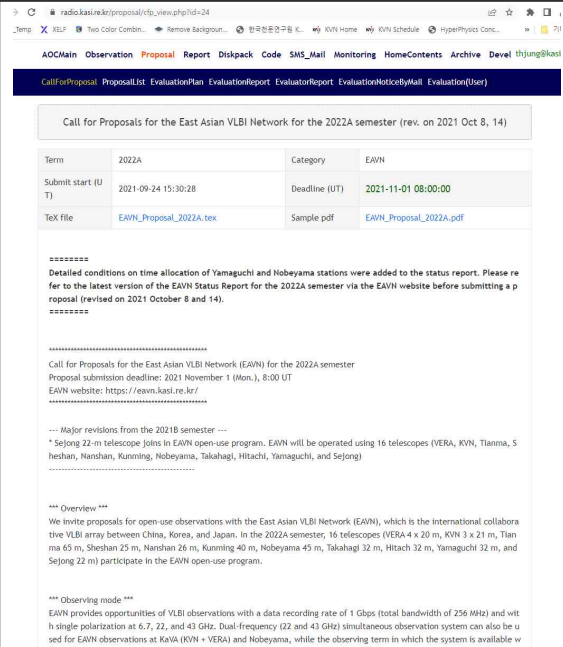
부록 6. 민간 연구자 지원 업무 협조

천문연구원에서 시행하고 있는 민간 연구자 지원 관련 업무협조 내용은 다음과 같다.

1. KVN/EAVN 민간 연구자 지원을 위한 관측제안서 모집 공고에 세종 우주측지관측센터 전파망원경을 포함하여 공고하였으며, 민간 연구자 제안서 모집, 심사 과정을 지원함

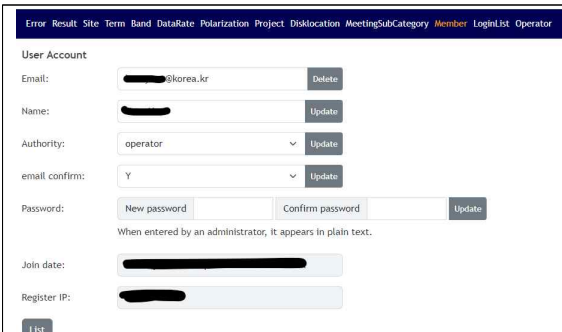


KVN 민간 연구자 관측제안서 모집 공고에 게재된 세종 전파망원경 활용



EAVN 민간 연구자 관측제안서 모집 공고에 게재된 세종 전파망원경 활용

2. 우주측지관측센터 전파망원경이 KVN/EAVN과 공동 관측을 수행하기 위한 웹 기반의 AOC(Array Operation Center; 관측망 운영 센터) 계정 발급 및 세종 관련 DB 생성



AOC 멤버리스트에 세종 담당자 추가



AOC에 구축되어있는 세종(KSJ) DB

3. KVN/EAVN 관측운업을 위한 AOC 활용법 교육
4. 민간 연구자 지원을 위한 EAVN 정기월례회의(Regular Meeting)과 소장회의(Director's Meeting) 참여

부록 7. EAVN 협력에 관한 연구논문 출판

세종 연구자가 함께 참여한 EAVN 연구논문 출판 (저널명: Galaxies, 2022년)



Review

Overview of the Observing System and Initial Scientific Accomplishments of the East Asian VLBI Network (EAVN)

Kazunori Akiyama^{1,2,3}, Juan-Carlos Algaba⁴, Tao An^{5,6}, Keiichi Asada⁷, Kitiyanee Asanok⁸, Do-Young Byun^{9,10}, Thanapol Chanapote⁸, Wen Chen¹¹, Zhong Chen⁵, Xiaopeng Cheng⁹, James O. Chibueze^{12,13}, Ilje Cho¹⁴, Se-Hyung Cho^{9,15}, Hyun-Soo Chung⁹, Lang Cui^{6,16}, Yuzhu Cui¹⁷, Akihiro Doi^{18,19}, Jian Dong⁵, Kenta Fujisawa^{20,21}, Wei Gou⁵, Wen Guo⁵, Kazuhiro Hada^{2,22}, Yoshiaki Hagiwara^{23,24,25}, Tomoya Hirota^{2,22}, Jeffrey A. Hodgson²⁶, Mareki Honma^{2,22,27}, Hiroshi Imai^{28,29}, Phrudth Jaroenjittichai⁸, Wu Jiang^{5,6}, Yongbin Jiang⁵, Yongchen Jiang⁵, Takaaki Jike^{2,22}, Dong-Kyu Jung⁹, Taehyun Jung⁹, Noriyuki Kawaguchi², Dong-Jin Kim^{1,30}, Hyo-Ryoung Kim⁹, Jaeheon Kim⁹, Jeong-Sook Kim^{31,32}, Kee-Tae Kim^{9,10}, Soon-Wook Kim⁹, Motoki Kino^{2,33}, Hideyuki Kobayashi², Shoko Koyama^{7,34}, Busaba H. Kramer^{8,30}, Jee-Won Lee⁹, Jeong Ae Lee⁹, Sang-Sung Lee^{9,10}, Sang Won Lee³⁵, Bin Li⁵, Guanghui Li¹⁶, Xiaofei Li¹⁶, Zhixuan Li¹¹, Qinghui Liu⁵, Xiang Liu^{6,16}, Ru-Sen Lu^{5,6,30}, Kazuhito Motogi^{20,21}, Masanori Nakamura^{7,36}, Kotaro Niinuma^{20,21}, Chungsik Oh⁹, Hongjong Oh³⁵, Junghwan Oh^{25,26}, Se-Jin Oh⁹, Tomoaki Oyama², Jongho Park^{7,9}, Saran Poshychinda⁸, Hyunwook Ro^{9,37}, Duk-Gyoo Roh⁹, Wiphu Rujopakarn⁸, Nobuyuki Sakai⁸, Satoko Sawada-Satoh³⁸, Zhi-Qiang Shen^{5,6}, Katsunori M. Shibata², Bong Won Sohn^{9,10,37}, Boonrucksar Soonthornthum⁸, Koichiro Sugiyama⁸, Yunxia Sun⁵, Mieko Takamura^{2,27}, Yoshihiro Tanabe³⁹, Fumie Tazaki⁴⁰, Sascha Trippe¹⁵, Kiyoaki Wajima^{9,10,41}, Jinqing Wang⁵, Na Wang^{6,16}, Shiqiang Wang¹⁶, Xuezheng Wang⁵, Bo Xia⁵, Shuangjing Xu^{5,9}, Hao Yan¹⁶, Wenjun Yang¹⁶, Jae-Hwan Yeom⁹, Kunwoo Yi¹⁵, Sang-Oh Yi³⁵, Yoshinori Yonekura³⁹, Hasu Yoon³⁵, Linfeng Yu⁵, Jianping Yuan¹⁶, Youngjoo Yun⁹, Bo Zhang⁵, Hua Zhang¹⁶, Yingkang Zhang⁵, Guang-Yao Zhao¹⁴, Rongbing Zhao⁵, Weiye Zhong⁵ and on behalf of the East Asian VLBI Network Collaboration[†]



Citation: Akiyama, K.; Algaba, J.-C.; An, T.; Asada, K.; Asanok, K.; Byun, D.-Y.; Chanapote, T.; Chen, W.; Chen, Z.; Cheng, X.; et al. Overview of the Observing System and Initial Scientific Accomplishments of the East Asian VLBI Network (EAVN). *Galaxies* **2022**, *10*, 113. <https://doi.org/10.3390/galaxies10060113>

Academic Editor: Luigina Ferretti

Received: 27 September 2022

Accepted: 29 November 2022

Published: 7 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

- ¹ Massachusetts Institute of Technology Haystack Observatory, Westford, MA 01886, USA
- ² Mizusawa VLBI Observatory, National Astronomical Observatory of Japan, Iwate 023-0861, Japan
- ³ Black Hole Initiative, Harvard University, Cambridge, MA 02138, USA
- ⁴ Department of Physics, Faculty of Science, Universiti Malaya, Kuala Lumpur 50603, Malaysia
- ⁵ Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai 200030, China
- ⁶ Key Laboratory of Radio Astronomy, Chinese Academy of Sciences, Nanjing 210008, China
- ⁷ Institute of Astronomy and Astrophysics, Academia Sinica, Hilo, HI 96720, USA
- ⁸ National Astronomical Research Institute of Thailand (Public Organization), Chiangmai 50180, Thailand
- ⁹ Korea Astronomy and Space Science Institute, Yuseong-gu, Daejeon 34055, Republic of Korea
- ¹⁰ Department of Astronomy and Space Science, University of Science and Technology, Yuseong-gu, Daejeon 34113, Republic of Korea
- ¹¹ Yunnan Observatories, Chinese Academy of Sciences, Kunming 650011, China
- ¹² Centre for Space Research, North-West University, Potchefstroom 2520, South Africa
- ¹³ Department of Physics and Astronomy, Faculty of Physical Sciences, University of Nigeria, Nsukka 410001, Nigeria
- ¹⁴ Instituto de Astrofísica de Andalucía-CSIC, Glorieta de la Astronomía s/n, E-18008 Granada, Spain
- ¹⁵ Department of Physics and Astronomy, Seoul National University, Gwanak-gu, Seoul 08826, Republic of Korea
- ¹⁶ Xinjiang Astronomical Observatory, Chinese Academy of Sciences, Urumqi 830011, China
- ¹⁷ Tsung-Dao Lee Institute, Shanghai Jiao Tong University, Shanghai 201210, China
- ¹⁸ The Institute of Space and Astronautical Science, Japan Aerospace Exploration Agency, Kanagawa 252-5210, Japan
- ¹⁹ Department of Space and Astronautical Science, The Graduate University for Advanced Studies, SOKENDAI, Kanagawa 252-5210, Japan
- ²⁰ The Research Institute for Time Studies, Yamaguchi University, Yamaguchi 753-8511, Japan
- ²¹ Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Yamaguchi 753-8512, Japan
- ²² Department of Astronomical Science, The Graduate University for Advanced Studies, SOKENDAI, Tokyo 181-8588, Japan
- ²³ Natural Science Laboratory, Toyo University, Bunkyo-ku, Tokyo 112-8606, Japan
- ²⁴ ASTRON, 7991 PD Dwingeloo, The Netherlands
- ²⁵ Joint Institute for VLBI ERIC, 7991 PD Dwingeloo, The Netherlands
- ²⁶ Department of Physics and Astronomy, Sejong University, Gwangjin-gu, Seoul 05006, Republic of Korea
- ²⁷ Institute of Astronomy, The University of Tokyo, Tokyo 181-0015, Japan

Abstract

Research and Technical Development for the VLBI Data Analysis Acceleration [2nd]

Taehyun Jung, Do-Heung Je, Do-Young Byun, Hyo Ryoung Kim, Jaeheon Kim

Research and technical development for the VLBI data analysis acceleration includes a wide range of research ranging from VLBI observations, correlation process and system performance improvement in order to accelerate VLBI data analysis of the Sejong's 22m radio telescope at the Sejong Space Geodetic Center. Compatibility with other VLBI telescopes is a very important factor due to the nature of VLBI observations, in which geodetic performance is highly dependent on the number of telescopes participating in the geodetic observations. For this, Sejong's backend system, DBBC3, was upgraded from 2L2H to 4L4H by equipping two more sampling board and its optimization and performance measurements were completed. A frequency synthesizer tailored to the input frequency band of DBBC3 was developed. As a result, Sejong is able to acquire VLBI data of four broadband signals in the S/X and K/Q bands. The generated VLBI data is processed using the Sejong's DiFX correlator, and procedures and related utilities were developed accordingly with incidental utility programs to automate time delay error correction, to manage VLBI data storage, and to increase correlation processing speed.

The performance of the DiFX software correlator was optimized after upgrading the Sejong's linux cluster, whose performance was improved by adding three computing nodes. Its environment and firewall setup were also revised. As a result, 2Gbps e-VLBI observation and realtime correlation of four radio telescopes of Sejong and the KVN (Korean VLBI Network) were successfully performed. Relevant research fields such as space geodesy, space exploration and near field VLBI are beneficial. In 2021, a memorandum of understanding (MOU) was signed with the East Asia VLBI Observation Network (EAVN) institutions, in which seven institutions from four countries, on cooperation in joint observation and operation, establishing the basis for the VLBI collaborations. In order to share the telescope performance information for international VLBI observations, the status report of Sejong 22m radio telescope was presented.

Through the 1st and 2nd research and development of VLBI Data Analysis Acceleration, Sejong VLBI telescope is able to provide better quality of VLBI data and faster correlation results. Based on these aspects, Sejong will be able to produce research outcomes in the various fields such as geodesy, astronomy, and space missions.

주 의 사 항

1. 본 보고서는 국토교통부 국토지리정보원의 수탁을 받아 한국천문연구원에서 수행한 보고서입니다.
2. 본 내용을 대외적으로 게재, 인용할 때에는 반드시 국토교통부 국토지리정보원의 사전 허락을 받기 바라며, 무단 복제는 절대 금합니다.

VLBI 데이터 분석 가속화 기술개발 연구[2차]

인쇄·2022년 12월

발행·2022년 12월

발행자·조 우 석

발행처·국토교통부 국토지리정보원

주소·경기도 수원시 영통구 월드컵로 92(원천동)

전화·031-210-2700

FAX·031-210-2644