

# LTE 관련 정리

ICT 융합학부  
이수진

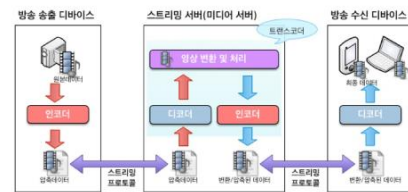
## 1. 영상 처리

1) Scalable Coding 을 이용하여 드론에서 촬영하는 영상을 딜레이 없이 스트리밍하기

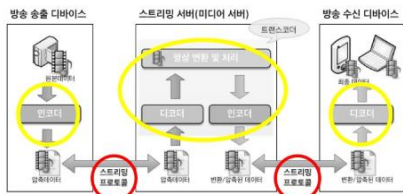
Scalable Coding을 이용하여  
드론에서 촬영한 영상을  
실시간으로 딜레이 없이 전송하기

ICT 융합학부  
이수진

### 영상 스트리밍 과정



### 영상 스트리밍 과정

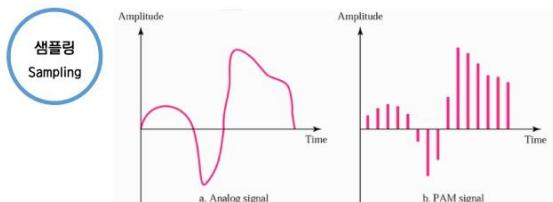


### 영상 스트리밍 과정

Coder : 아날로그 신호 -> 디지털 신호

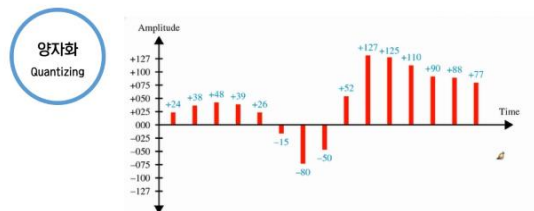


### 영상 스트리밍 과정



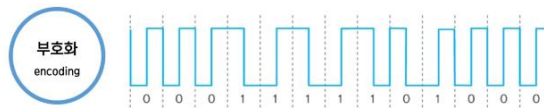
Nyquist 이론  
최소한 최대 주파수의 2배만큼은 샘플링을 해야 수신 측에서 원래 신호  
를 복원 할 수 있다

### 영상 스트리밍 과정



PAM 시그널 -> 정수 레벨

### 영상 스트리밍 과정



양자화된 신호 -> 디지털 영상 신호

“압축”

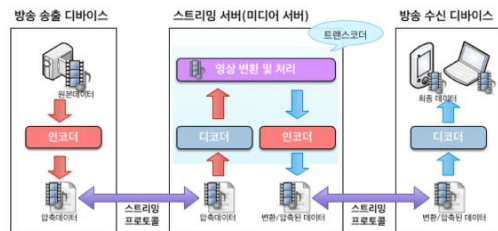
## 드론에서 영상 전송

ICT 미수진

## Contents

- 01 영상 전송 과정
- 02 드론
- 03 latency

영상 전송 과정

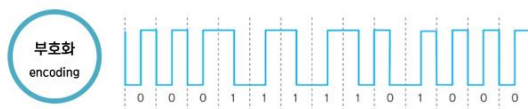


영상 전송 과정

인코딩 : 아날로그 신호 -&gt; 디지털 신호



영상 전송 과정



양자화된 신호 -&gt; 디지털 영상 신호

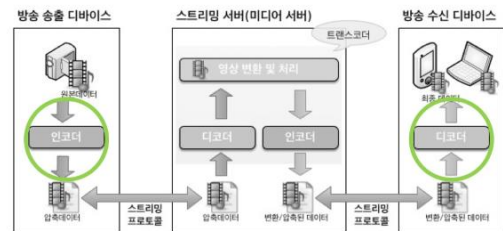
“압축”

드론



“사용자에 의한 움직임”

영상 전송 과정



영상 전송 과정

인코딩 : 아날로그 신호 -&gt; 디지털 신호



영상 전송 과정

“압축” = 예측 + 변환

latency

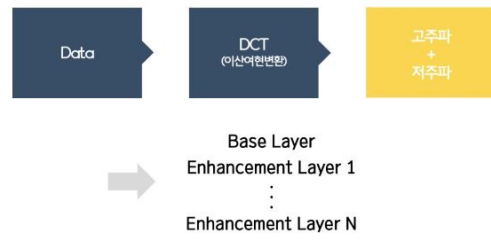
움직임 보상 프레임 간 예측

드론이 사용자의 조종에 의해 움직이는  
특성상 연속적인 장면이 나오게 됨

latency

# DCT & Scalable Coding

latency



latency

Base Layer  
Enhancement Layer 1  
⋮  
Enhancement Layer N



우선 순위에 따라  
디코딩

latency

## DCT

소요시간 ↑  
-> 움직이지 않을 때 사용 x

## 2. NB-IoT survey 및 Particle

### 1) Survey

#### NB-IoT

ICT 융합학부 이수진

#### A Primer on 3GPP Narrowband Internet of Things

Y.-P. Eric Wang, Xingqin Lin, Ansuman Adhikary, Asnjorn Grovlen ...

#### Introduction

NB-IoT aims to offer deployment flexibility, allowing an operator to introduce NB-IoT using a small portion of its existing available spectrum.

#### Transmission Schemes and Deployment Options

##### Downlink Transmission Scheme

Based on OFDMA(15kHz)

->LTE ensures good coexistence performance with LTE in the downlink

#### Transmission Schemes and Deployment Options

##### Deployment Options

NB-IoT may be deployed as a standalone carrier.

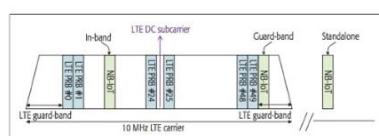


Figure 1. Examples of NB-IoT stand-alone deployment and LTE in-band and guard-band deployments.

#### NB-IoT

#### Narrowband - Internet of Things

협대역 사물 인터넷

#### Abstract

Narrowband-Internet of Things (NB-IoT)

New cellular technology introduced in 3GPP Release 13

Overview of the air interface of NB-IoT

#### Introduction

State-of-the-art overview  
of the air interface of NB-IoT with a focus  
on the key aspects where NB-IoT deviates from LTE.

#### Transmission Schemes and Deployment Options

##### Uplink Transmission Scheme

Multi-tone trans. :: Based on single - carrier frequency-division multiple access using the same 15kHz subcarrier spacing and 0.5 ms slot at LTE.

single-tone trans. :: Support two numerologies, 15kHz and 3.75kHz

#### Physical Channels

##### Downlink

Unlike LTE, these NB-IoT physical channels and signals are primarily multiplexed in time.

	Subframe number									
	0	1	2	3	4	5	6	7	8	9
Even numbered frame	NPBCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPSS	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NSS
Odd numbered frame	Subframe number									
	0	1	2	3	4	5	6	7	8	9
	NPBCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPSS	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH	NPOCCH or NPDSCH

Figure 2. Time multiplexing of NB-IoT downlink physical channels and signals.

## Physical Channels

### Uplink

NPRACH :: is a newly designed channel

Since the legacy LTE physical random access channel uses a bandwidth of 1.08MHz, more than NB-IoT uplink bandwidth.

NPUSCH :: has two formats

Format 1 is used for carrying uplink data and uses the same LTE turbo code for error correction.

Format 2 is used for signaling HARQ acknowledgment for NPDSCH, and uses a repetition code for error correction.

## Cell Search and Initial Acquisition Procedure

UE needs to distinguish a particular cell on the basis of an NB-PCID

NB-IoT provide extended coverage for UEs deployed in environments.

Despite this large CFO, a UE should also be able to perform accurate synchronization at very low SNR(Signal to noise ratio).

Synchronization process principles are similar to LTE .

## Scheduling and HARQ operation

NB-IoT allows only one HARQ process in both downlink and uplink, and allows longer UE decoding time for both NPDCCH and NPDSCH.

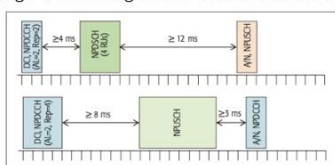


Figure 5. Timing relationship operation (each unit corresponds to one sub-frame).

## Conclusion

In this article, a description of NB-IoT radio access is given.

NB-IoT ushers in ultra-low-cost devices and has enough capacity to support a massive number of these devices in a cell.

## Resource Mapping

The orthogonality to LTE signals is preserved by avoiding mapping NB-IoT signals to the resource elements already used by the legacy LTE signals.

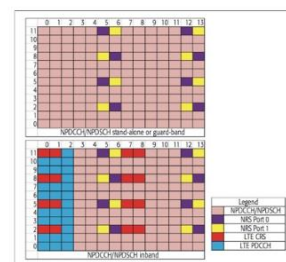


Figure 4. NPDCCH/NPDSCH resource mapping example.

## Random Access

In NB-IoT, random access is to achieve uplink synchronization.

NB-IoT allows flexible configuration of NPRACH resources in a time0frequency resource grid.

UE determines its coverage level by measuring downlink received signal power.

## Performance

Peak Data Rates

Coverage

Device Complexity

Latency and battery lifetime

Capacity

## 2) Particle 주문내역



Particle Industries, Inc.  
126 Post St 4th Floor  
San Francisco CA 94108  
United States

### Bill To

Sujin Lee  
55, Hanyangdaehak-ro, Sangnok-gu  
601, ERICA center (Education Res...  
Ansan-si Gyeonggi-do 15588  
Korea, Republic of

### Ship To

Sujin Lee  
55, Hanyangdaehak-ro, Sangnok-gu  
601, ERICA center (Education Res...  
Ansan-si Gyeonggi-do 15588  
Korea, Republic of

## Sales Order

Date 11/2/2018  
Order # SO20035  
  
Terms Due on receipt  
PO # MP6186  
Ship Via FedEx International Pri...  
Subsidiary 100 - Particle Industrie...  
Ship Date 11/5/2018  
Tracking # 472661459505  
Project

Item	Quantity	Unit	Serial/Lot Numbers	Description	Unit Price	Options	Amount	Tax R...	Item ID	Item Desc...
Adahut OLED FeatherWin B	1			Adahut OLED FeatherWing	14.95		14.95		FWNG-OL...	Adahut O...
Adahut RPA219 FeatherWin B	1			Adahut RPA219 Featherwing	7.95		7.95		FWNG-INL...	Adahut I...
Adahut Power Relay FeatherWin B	1			Adahut Power Relay FeatherWing	9.95		9.95		FWNG-P...	Adahut P...
ArgonR2	1			Argon R2	15.00		15.00		ARGONR2...	Argon R2
Boron 2G/3G R1	1			Boron 2G/3G R2	49.00		49.00		BRND19R...	Boron 2G...
BoronLTE R2	1			Boron LTE R1	29.00		29.00		BRND29R...	Boron LTE
Particle Classic Adapter	1			Particle Classic Adapter	5.00		5.00		ACC-ADPT	Particle Cl...
Particle Debugger	1			Particle Debugger	20.00		20.00		ACC-DEB...	Particle D...
Particle Ethernet FeatherWin B	1			Particle Ethernet FeatherWing	20.00		20.00		FWNG-ETH	Particle ET...
Grove StarterKit for Particle Modu	1				29.00		29.00		SNSR-GRL...	Grove Sta...
XenonR2	1			Xenon R1	9.00		9.00		XENRNT...	Xenon R2
Shipping and Handling Fees	1				15.00		15.00		SAH	Shipping...

**Total \$223.85**

### 3. LTE survey

## LTE survey

이수진

RSSI, RSRQ, RSRP

### RSSI

#### Reference Signal Strength Indicator

RSSI는 단말에 수신되는 Power의 총 크기이다. (절대적인 값)  
채널의 간섭, 열 잡음 등 모든 정보를 포함한다.  
RSSI는 N개의resource block에 대해서 측정한다.

### RSRP

#### Reference Signal Received Power

RSRP는 광대역/협대역에서 측정된 LTE Reference Signal의 수신 전력이다. (절대적인 값)  
단말이 보고하는 RSRP 범위는 -44 ~ -140 dBm 이다.  
통신 가능한 RSRP의 범위는 셀 중심부에서 -75dBm 정도이고 셀 엣지 부분에서는 -120dBm 정도이다.  
RSRP의 범위는 -140dBm ~ -44dBm으로 1dB 단위로 변경될 수 있다.

Reported value	Measured quantity value	Unit
RSRP_00	RSRP < -140	dBm
RSRP_01	-140 ≤ RSRP < -139	dBm
RSRP_02	-139 ≤ RSRP < -138	dBm
...	...	...
RSRP_95	-46 ≤ RSRP < -45	dBm
RSRP_96	-45 ≤ RSRP < -44	dBm
RSRP_97	-44 ≤ RSRP	dBm

네트워크 품질 매핑 표

### RSRQ

#### Reference Signal Received Quality

RSRQ는 단말에 수신되는 Power 대비 Reference Signal Power의 비다.  
RSSI와 사용된 resource block의 수와 간섭등 다양한 정보를 고려한다.  
 $RSRQ = (N \cdot RSRP) / RSSI$  (N: RB의 수)  
단말이 보고하는 RSRQ의 범위는 -3dB ~ -19dB이다.

Reported value	Measured quantity value	Unit
RSRQ_00	RSRQ < -19.5	dB
RSRQ_01	-19.5 ≤ RSRQ < -19	dB
RSRQ_02	-19 ≤ RSRQ < -18.5	dB
...	...	...
RSRQ_32	-4 ≤ RSRQ < -3.5	dB
RSRQ_33	-3.5 ≤ RSRQ < -3	dB
RSRQ_34	-3 ≤ RSRQ	dB

네트워크 품질 매핑 표

## 기지국 명령어

HARQ 최대 전송 가능 횟수

ITBS 값 고정

CQI 값 고정

Resource Block 할당 최대치 조절

### HARQ Max Tx Override

JL620(config-engineer)# set hyu harq downlink <1-4>

HARQ DL 최대 전송 가능 횟수를 지정.  
기본값은 4임.

JL620(config-engineer)# set hyu harq uplink <1-4>

HARQ UL 최대 전송 가능 횟수를 지정.  
기본값은 4임.

### System-wide ITBS Override

JL620(config-engineer)# set hyu itbs-override downlink <0-100>

Downlink ITBS 값을 임의로 고정.

JL620(config-engineer)# set hyu itbs-override uplink <0-100>

Uplink ITBS 값을 임의로 고정.

### UE-specific ITBS Override

JL620(config-engineer)# set hyu itbs-override downlink <0-100> crnti <1-200>

Downlink ITBS 값을 특정 기기에 대해 고정

JL620(config-engineer)# set hyu itbs-override uplink <0-100> crnti <1-200>

Uplink ITBS 값을 특정 기기에 대해 고정

### System-wide CQI Override

JL620(config-engineer)# set hyu cqi-override downlink <0-15>

단말이 보고하는 Downlink CQI 값을 임의로 값으로 고정.

JL620(config-engineer)# set hyu cqi-override uplink <0-15>

단말이 보고하는 Uplink CQI 값을 임의로 값으로 고정.

\*System Wide ITBS보다 우선순위가 높음

System-wide Max Resource Block Limits

```
JL620(config-engineer)# set hyu rb-override downlink <0-100>
```

RB 사용량을 제한.  
20MHz 채널에서 최대 100개까지의 RB 할당 가능하나, 이 최대치를 임의로 조절.

```
JL620(config-engineer)# set hyu rb-override uplink <0-100>
```

RB 사용량을 제한.  
20MHz 채널에서 최대 100개까지의 RB 할당 가능하나, 이 최대치를 임의로 조절.

CellSignalStrengthLte

CellSignalStrengthLte

```
public final class CellSignalStrengthLte
extends CellSignalStrength implements Parcelable

java.lang.Object
└─ android.telephony.CellSignalStrength
    └─ android.telephony.CellSignalStrengthLte
```

CellSignalStrengthLte

method	Return value
getAsuLevel()	Get the LTE signal level as an asu value between 0.97, 99 is unknown Asu is calculated based on 3GPP RSRP.
getCqi()	Get channel quality indicator
getDbm()	Get signal strength as dBm
getLevel()	Get signal level as an int from 0.4
getRsrp()	Get reference signal received power
getRsrq()	Get reference signal received quality
getRssnr()	Get reference signal signal-to-noise ratio
getTimingAdvance()	Get the timing advance value for LTE, as a value in range of 0.1282.

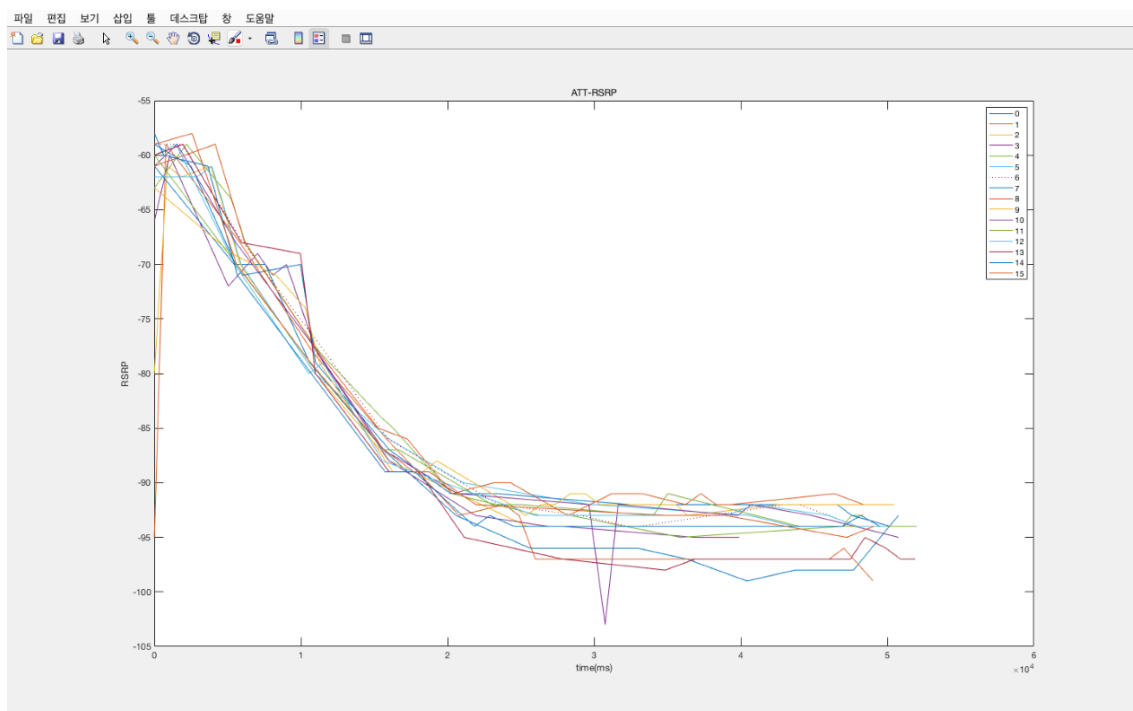


## 4. LTE 실험 진행

### \* 간섭에 따른 RSRP 값

간섭에 따라서 RSRP 값이 어떻게 변화하는지 알아야했다. 처음으로 한 것은 RSRP 값을 찍는 어플리케이션을 만드는 것이었다. 기지국의 HARQ 값은 4로 고정하고, CQI 값을 0부터 15까지 1씩 증가시켰다. 각 CQI 값에 따라서 attenuator 값을 5초에 10dB씩 증가시키면서 실험을 진행했다. 그 결과 attenuator로 인한 간섭이 클 수록 RSRP 값이 낮아진다는 사실을 확인할 수 있었다.

### 1) 실험 결과 그래프



실험 결과

### 2) Shell script code

```
#!/bin/bash
sudo j1sh <<"EOF"
conf engineer
set hyu harq downlink 3
set hyu harq uplink 3
EOF

i=1

echo "-----CQI = 0-----"
sudo j1sh <<"EOF"
conf engineer
set hyu set cqj-override downlink 0
```

```
set hyu set cqi-override uplink 0
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 1—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 1
set hyu set cqi-override uplink 1
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 2—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 2
set hyu set cqi-override uplink 2
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 3—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 3
set hyu set cqi-override uplink 3
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 4—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 4
set hyu set cqi-override uplink 4
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 5—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 5
set hyu set cqi-override uplink 5
exit
exit
EOF
```

```
sleep $i
```

```
echo "—————CQI = 6—————"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 6
set hyu set cqi-override uplink 6
```

```
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 7_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 7
set hyu set cqi-override uplink 7
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 8_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 8
set hyu set cqi-override uplink 8
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 9_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 9
set hyu set cqi-override uplink 9
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 10_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 10
set hyu set cqi-override uplink 10
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 11_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 11
set hyu set cqi-override uplink 11
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 12_____"
sudo jlsh <<"EOF"
conf engineer
set hyu set cqi-override downlink 12
set hyu set cqi-override uplink 12
```

```
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 13_____"
sudo j1sh <<"EOF"
conf engineer
set hyu set cqi-override downlink 13
set hyu set cqi-override uplink 13
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 14_____"
sudo j1sh <<"EOF"
conf engineer
set hyu set cqi-override downlink 14
set hyu set cqi-override uplink 14
exit
exit
EOF
```

```
sleep $i
```

```
echo "_____CQI = 15_____"
sudo j1sh <<"EOF"
conf engineer
set hyu set cqi-override downlink 15
set hyu set cqi-override uplink 15
exit
exit
EOF
```

```
sleep $i
```

### 3) Attenuator

```
sc set 0 0 90 10 5000 10000 2 0
```

(type, start, end, step, interval, hold, counter, direction)

### 4) Application

어플리케이션은 총 두 개를 만들었다. 첫 번째는 signal strength 가 변화하면 RSRP 값이 기록이 되며 txt 파일에 저장이 된다. 버튼을 누르면 기록이 시작되며 기록 시작 시각과 signal strength 가 변하면 경과한 시간, RSRP 값이 기록이 된다. 이는 PhoneStateListener 클래스를 이용하여 만들었으며 단말기의 API 값이 1 이상이면 사용할 수 있다. 두 번째는 AsuLevel, CQI, Dbm, signal level, RSRP, RSRQ, signal-to-noise ratio 그리고 timing advance value for LTE 값을 얻을 수 있다. 이는 API 26 부터 사용할 수 있다.

## 5) 기록 예시

	A	B	C	D	E	F	G	H	I	J
1	AsuLevel	CQI	dBm	level	RSRP	RSRQ	RSSNR	timing advanced		
2	0 #####									
3	4758	42	2147483647	-98	3	-98	-6	20	8	
4	19680	45	2147483647	-95	3	-95	-5	26	8	
5	42754	40	2147483647	-100	3	-100	-6	11	8	
6	83721	41	2147483647	-99	3	-99	-8	20	8	
7	87546	41	2147483647	-99	3	-99	-6	11	8	
8	155381	42	2147483647	-98	3	-98	-6	22	8	
9	163065	41	2147483647	-99	3	-99	-5	21	8	
10	544509	41	2147483647	-99	3	-99	-7	23	8	
11	0 #####									
12	11168	41	2147483647	-99	3	-99	-6	23	8	
13	15350	41	2147483647	-99	3	-99	-7	21	8	
14	19188	44	2147483647	-96	3	-96	-7	26	8	
15	21741	46	2147483647	-94	3	-94	-7	26	8	
16	0 #####									
17	22783	42	2147483647	-98	3	-98	-7	15	8	
18	24441	42	2147483647	-98	3	-98	-7	15	8	
19	32155	42	2147483647	-98	3	-98	-6	14	8	
20	39820	43	2147483647	-97	3	-97	-7	14	8	
21	44956	43	2147483647	-97	3	-97	-5	13	8	
22	48795	43	2147483647	-97	3	-97	-7	14	8	
23	53915	43	2147483647	-97	3	-97	-5	22	8	

## \*발표자료

# URP 발표

ICT 융합학부 이수진

2019.08.23

## 목차

1. 연구 주제  
; 연구 주제 및 목표
2. 실험 결과  
; 연구 결과 및 차이, 데이터
3. 연구 확장  
; 연구 확장 계획
4. 소감  
; 한 학기 동안 느낀 점

## 1. 연구 주제

# LTE Link Adaptation

; 데이터의 특성에 따라서 효율적으로 전송

## 1. 연구 주제

# LTE Link Adaptation

- 링크 적응 기술로 무선 채널 상태의 변화에 따라 전송 파라미터를 적응 조절하는 기술로  
전송속도, 시스템 수용 용량 및 주파수 효율을 높이는 데 목적이 있음

[출처] 정보통신 기술 용어 백과사전

## 1. 연구 주제; 연구 과정

무선 통신 및 영상  
처리 관련 공부하기

통신 환경에 따른  
LTE signal strength와  
error rate 실험

Scalable Coding을  
통한 데이터 전송 효율  
향상 방법 연구

## 2. 실험 결과; 실험 환경





기저국

Attenuator  
(감쇠기)

EPC

## 2. 실험 결과; 용어

### CQI :: Channel Quality Indicator

이동단말에 의해 무선 채널의 품질을 측정해 하고, 주기적으로 이러한 채널 품질 정보를 기지국에 전달하는 수평링크 부속임/정보를 말함

### HARQ :: Hybrid Automatic Repeat Request

열악한 무선 채널 환경 등으로 인해 자주 발생하는 단말측에서 기지국으로의 재전송 요구를 줄여 데이터 전송 효율을 높이는 방식

### RSRP :: Reference Signal Received Power

열악한 무선 채널 환경 등으로 인해 자주 발생하는 단말측에서 기지국으로의 재전송 요구를 줄여 데이터 전송 효율을 높이는 방식

## 2. 실험 결과; ITBS Override



Base Station

HARQ 값 3으로 고정  
CQI 값을 0에서부터 15까지 1씩 증가



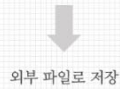
Attenuator

0dB부터 190dB까지 10dB씩 증가

CQI와 attenuator 값을  
시간에 따라 변경

## 2. 실험 결과; ITBS Override

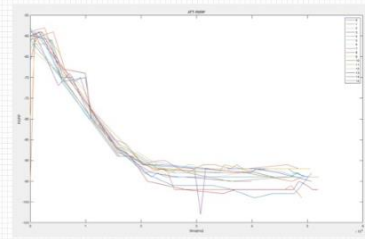
단말기에서 어플리케이션을 통해  
**LTE signal strength** 측정  
(RSRP, RSRQ, RSSNR, CQI, Asu Level, Signal strength at Dbm)



외부 파일로 저장



## 2. 실험 결과; ITBS Override



## 2. 실험 결과; HARQ - Error Rate



Base Station

HARQ 값 1부터 4까지 변경  
(재전송 횟수 0부터 3)



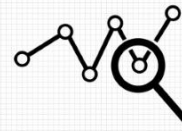
Server

라즈베리를 이용하여 서버 구축



재전송 횟수에 따른  
Error rate 측정

## 3. 연구 확장;



수집한 데이터를 바탕으로 통신 환경에 따라서  
지연 시간을 최소화 할 수 있는 방법을 찾는 방향으로 확장

## 4. 소감;



감사합니다

## 5. RTSP 서버 파일 전송 실험

### 1) 실험 환경

- 'Cherry Music'

오픈소스를 바탕으로 라즈베리파이를 이용해 음악 스트리밍을 할 수 있는 RTSP 서버 구축  
(166.104.185.64:8088)

- Attenuator : P1 90 P2 90 P3 0 P4 0

기지국이 가까워서 그런지 너무 error 가 적게 발생하여 실험 환경을 좋지 않게 설정함

- 기지국 : HARQ Max Tx Override

HARQ DL(dir=0) 및 UL(dir=1)의 최대 전송 가능 횟수를 0 회, 1 회, 2 회, 3 회로 변경하여 실험 진행

- 음악 : 비투비 - 그리워하다 (3m 56sec)

### 2) 실험

- HARQ Tx=1 : 25.73 sec , 10.90%

- HARQ Tx=2 : 21.53 sec , 9.12%

- HARQ Tx=3 : 16.48 sec , 6.98%

- HARQ Tx=4 : 14.56 sec , 6.17%

### 3) 결론

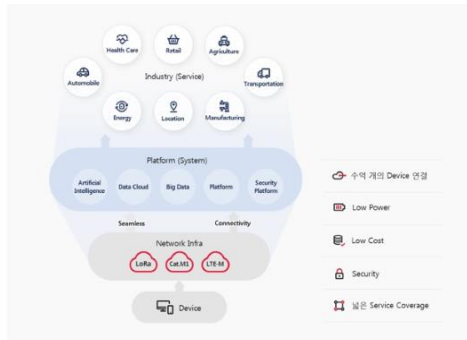
- HARQ 의 최대 재전송 횟수를 감소시킴에 따라서 error rate 이 높게 측정이 됨을 알 수 있음

- Attenuator 의 값을 90 으로 설정한 만큼 RSRP 값이 -100 정도로 측정되었다. 따라서 error rate 이 높게 측정되었음을 알 수 있다.

## 6. SK IoT survey

## SK IoT Survey

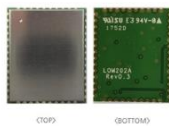
ICT융합학부  
이수진



## LORA LOM202A



TYPE	EMBEDDED SMD TYPE
Freq. Range	902.35MHz ~ 927.5MHz
Download Link	MAX DL 5.4 kbps / UL 5.4kbps



Product Features	
<ul style="list-style-type: none"> <li>• LoRa WAN Compliant</li> <li>• 128GB Internal Flash Memory</li> <li>• External ANT support</li> </ul>	<ul style="list-style-type: none"> <li>• Supply Voltage Typ: 3.3V</li> <li>• Standard and extended AT command set</li> <li>• Dimensions 25 x 38 x 2.2mm</li> </ul>
Interfaces	Approvals
<ul style="list-style-type: none"> <li>• 48-pin Embedded Interface</li> <li>• Various I/O ports including multifunctional I/Os</li> <li>• UART, SPI, etc.</li> </ul>	<ul style="list-style-type: none"> <li>• KC Certification</li> <li>• RoHS compliant</li> </ul>

## Cat.M1 Starter Kit



메인보드, RAM	10 x 10.5 mm
간헐작동용 보호 (무인작업 MM-M00055)	30 x 36 mm
간헐작동용 보호 (수동작업 MM-M00055)	35 x 40 mm
인쇄기	1000 x 12 mm
전원	50 x 100 mm, 50 mm (200W)

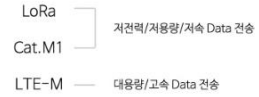
### Starter Kit 특징

- 부러진 ARM 칩을 쉽게 교체 가능
- Nano SIM & USB 지원
- 아무나도 쓰기 쉬운 보판
- 컴퓨터나 보드에서 동작 시 Push-Button으로 통신 연결 가능
- CAMS 모듈을 인기 전문 3P사의 CAMS 모듈 사용
- CAMS 모듈을 3P사의 모듈 없이 USB로만 작동할 수 있음
- PC를 통해 CAMS 모듈 AT Command 제어 가능 (USB)
- 다양한 개발 용품이 제공되고 제품 인증서, 워런티
- 2년 보증 서비스

## Cat.M1 Starter Kit

우리들의 IoT Starter Kit 구성품			AMN레노 IoT Starter Kit 구성품		
구분	부품명	수량	구분	부품명	수량
1.	MINI ARMV7F(Cortex-A7)	1	1.	ARMV7F(Cortex-A7)	1
2.	Starter Kit Base Board	1	2.	Starter Kit Base Board	1
3.	Interface Board	1	3.	Interface Board	1
4.	인전단 2개 (인전단 1개 포함)	1	4.	인전단 2개 (인전단 1개 포함)	1
5.	Micro USB 케이블	1	5.	Micro USB 케이블	1
6.	SV 전원 어댑터	1	6.	SV 전원 어댑터	1
7.	Nano IoT Card	1	7.	Nano IoT Card	1

## IoT Module



**LORA** LOM102A



TYPE	EMBEDDED SMD TYPE
Freq. Range	902.3MHz ~ 927.5MHz
Down/Up Link	MAX DL 5.4 kbps / UL 5.4 kbps



Product Features	
<ul style="list-style-type: none"> <li>• LoRa WAN Compliant</li> <li>• 540K Internal Flash Memory</li> <li>• External ANT support</li> </ul>	<ul style="list-style-type: none"> <li>• Supply Voltage Typ 3.3V</li> <li>• Standard and extended AT command set</li> <li>• Dimensions 17 x 26 x 2.65 mm</li> </ul>
Interfaces	Approvals
<ul style="list-style-type: none"> <li>• 48-pin Embedded Interface</li> <li>• Various I/O ports including multifunctional I/Os</li> <li>• UART, SPI, QMC etc.</li> </ul>	<ul style="list-style-type: none"> <li>• KC Certification</li> <li>• RoHS compliant</li> </ul>

**LORA** 이도링크

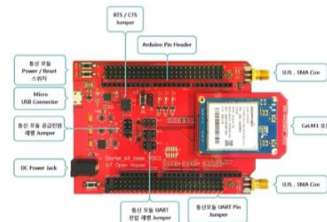
IM-L210/IM-L400은 Semtech사의 장거리 트랜시버 SX1276을 내장한 통신 모듈로 원격검침, 센서 네트워크 등의 IoT 응용에 적용할 수 있습니다.



SK telecom LoRa Module (IM-L210/IM-L400)		
Model	IM-L210 / IM-L400	
Freq. Range	917MHz ~ 923.5MHz	
Down/Up Link	MAXDL: 5.4 kbps / UL: 5 ddbps	

Product Features	
<ul style="list-style-type: none"> <li>- Size               <ul style="list-style-type: none"> <li>- RM-1210: 45mm X 20mm</li> <li>- RM-1400: 20mm X 16mm</li> </ul> </li> <li>- 일부분 용량을 선택 DIP type 2.0mm pitch header 가용</li> <li>- Power saving mode 제공</li> <li>- Interface               <ul style="list-style-type: none"> <li>- UART 통신 인터페이스 제공</li> <li>- 온보드 LED 상태 확인을 위한 GPIO 제공</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>- RF Transceiver               <ul style="list-style-type: none"> <li>- Ultra-low Power</li> <li>- IC 부하 없이 극대 전송</li> </ul> </li> <li>- Frequency Band: S17 9123.5MHz</li> <li>- 대역폭: 125KHz</li> <li>- Data Rate: S17 7가~5.4Kbps, S12 7가 29Kbps</li> <li>- Microcontroller               <ul style="list-style-type: none"> <li>- Ultra-low Power, High Performance</li> <li>- ARM Cortex-M3 32-bit CPU</li> <li>- 256KB In-System-programmable, 32KB RAM</li> </ul> </li> </ul>

## Cat.M1 Starter Kit



## Cat.M1 AM telecom



TYPE	PC3e M.2
Size/Weight	303 x 42 x 2.3
Bandwidth	17TB/s, 85
Data Speed	2x 3000ops/s, 16 3700ops
Interface	USB 3.0 SATA eSATA eSATA eSATA
Other Features	GPS Wi-Fi GSM Thunderbolt
Open SDK	X
Power Consumption	780
Operating Temperature	-40°C ~ +85°C
Certification	CE, FCC, RoHS



Cat.M1 우라넷 (LGA)\_WM-N400MS



TYPE	LGA
Size(mm)	26.7 x 26.2 x 2.05
Bandwidth	1.5Gbps
Data Speed	DL 300Mbps / UL 3750kps
Interface	UART USB 2.0 GSM GPRS
Other Features	VoLTE GPS PCMA SAR ThingPlug
Open SDK	Linux
Power Consumption	780
Operating Temperature	-25°C ~ +85°C
Certification	KC 인증됨

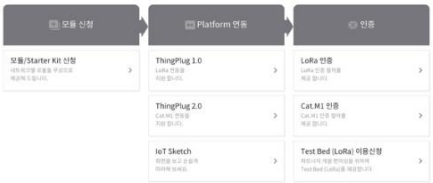
Cat.M1 우라넷 (LGA)\_WM-N400MSE



TYPE	Socket
Size(mm)	45.2 x 35 x 4.7
Bandwidth	1.5Gbps
Data Speed	DL 300Mbps / UL 3750kps
Interface	UART USB 2.0 GSM GPRS
Other Features	VoLTE GPS PCMA SAR ThingPlug
Open SDK	Linux
Power Consumption	780
Operating Temperature	-25°C ~ +85°C
Certification	KC 인증됨

Developer

개발 Process



LoRa / Cat.M1 전용 모듈 무상 대여

## 7. UDP 소켓 실험 진행

### 1) 코드

#### 가) Server code

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>

#define MAXLINE 511
#define BLOCK 255
#define FILENAME "buf.txt"

int main(int argc, char *argv[]) {
    struct sockaddr_in servaddr, cliaddr;
    int s, nbyte, addrlen = sizeof(struct sockaddr);
    char buf[MAXLINE+1];
    FILE *stream; //파일 입출력

    //파일명 포트번호
    if(argc != 2) {
        printf("usage: %s port\n", argv[0]);
        exit(0);
    }

    //소켓 생성
    if((s = socket(PF_INET, SOCK_DGRAM, 0)) < 0) {
        perror("socket fail");
        exit(0);
    }

    // 서버 구조
    memset(&cliaddr, 0, addrlen); //bzero((char *)&cliaddr, addrlen);
    memset(&servaddr, 0, addrlen); //bzero((char *)&servaddr, addrlen);
    servaddr.sin_family = AF_INET;
    servaddr.sin_addr.s_addr = htonl(INADDR_ANY);
    servaddr.sin_port = htons(atoi(argv[1])); //argv[1]에서 port 번호 가지고 옴

    // 서버 로컬 주소로 bind()
    if(bind(s, (struct sockaddr *)&servaddr, addrlen) < 0) {
        perror("bind fail");
        exit(0);
    }

    //저장용 파일 생성
    if((stream = fopen(FILENAME, "w")) == 0) {
        printf("Faile open error\n");
        exit(1);
    }
    while(1)
    {
```

```

    puts("Server : waiting request.");
    //전송 받은 메시지 nbyte 저장
    nbyte = recvfrom(s, buf, MAXLINE , 0, (struct sockaddr *)&cliaddr,
&addrlen);
    if(nbyte< 0) {
        perror("recvfrom fail");
        exit(1);
    }
    buf[nbyte] = 0; //마지막 값에 0

    if(!strncmp(buf, "end of file", 10)) { //마지막 메시지가 end of file 이면
종료
        printf("file close");
        fclose(stream); //stream 닫기
        break; //while 문 빠져나가기
    } else {
        printf("%d byte recv: %s\n",nbyte, buf);
        fputs(buf, stream); //파일로 저장
    }
    puts("sendto complete");
}
if((stream = fopen(FILENAME, "r")) == NULL) {
    printf("Read File Error");
    exit(1);
}

while(!feof(stream)) {
    buf[0] = '\0';
    fgets(buf, BLOCK, stream);
    printf("Send : %s\n", buf);
    //메시지 전송
    if(sendto(s, buf, strlen(buf), 0, (struct sockaddr *)&cliaddr,
addrlen) < 0) {
        perror("sendto fail");
        exit(0);
    }
}
fclose(stream);
close(s);
return 0;
}

```

#### 나) Application code

```

package com.example.leesujin.udp_socket;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

//UDP 관련

```

```

import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;

public class MainActivity extends AppCompatActivity {
    //서버주소
    //public static final String sIP = "192.168.0.124";
    //사용할 통신 포트
    public static final int sPORT = 8088;

    //데이터 보낼 클래스
    public SendData mSendData = null;

    //화면 표시용 TextView
    public TextView txtView = null;
    public TextView txtView2 = null;
    public int i=0;

    //보낼 문자열
    public String s_msg = "(문자열 입력)";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //btnHello 버튼을 layout의 버튼과 연결
        Button btnHello = (Button) findViewById(R.id.Hello);
        //txtView를 layout의 TextView와 연결
        txtView = (TextView) findViewById(R.id.textView);
        txtView2 = (TextView) findViewById(R.id.textView2);

        //버튼이 눌렀다면
        btnHello.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //SendData 클래스 생성
                mSendData = new SendData();
                //보내기 시작
                mSendData.start();
                i++;
            }
        });
    }

    //데이터 보내는 쓰레드 클래스
    class SendData extends Thread{
        public void run(){
            try{
                //UDP 통신용 소켓 생성

```

```

        DatagramSocket socket = new DatagramSocket();
        //서버 주소 변수
        InetAddress serverAddr =
InetAddress.getByName("166.104.185.64");

        Log.i("information",serverAddr.toString());

        //보낼 데이터 생성
        byte[] buf = (s_msg).getBytes();
        Log.i("process",buf.toString());

        //패킷으로 변경
        Log.i("process","packet");
        DatagramPacket packet = new DatagramPacket(buf, buf.length,
serverAddr, sPORT);

        //패킷 전송!
        Log.i("process","send");
        socket.send(packet);

        //데이터 수신 대기
        Log.i("process","waiting");
        socket.receive(packet);
        //데이터 수신되었다면 문자열로 변환
        String msg = new String(packet.getData());

        //textView에 표시
        Log.i("result",msg);
        textView2.setText(i+" time try error rate : "+(100-
(msg.length()/s_msg.length()*100))+"%");
        textView.setText(msg);
    }catch (Exception e){

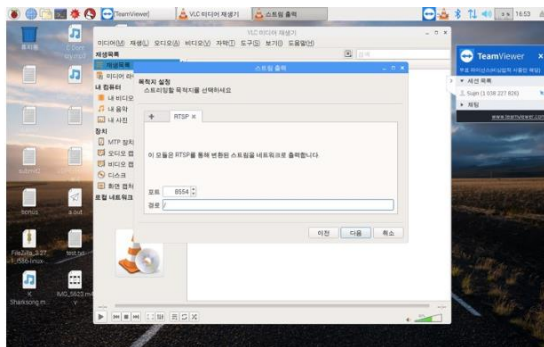
    }
}
}
}

```

## 8. HARQ 에 따른 error rate 측정

### LTE 실험 HARQ에 따른 Error rate 측정

ICT융합학부 이수진



### 실험

#### 목적

- LTE의 재전송의 횟수가 감소하게 되면 error rate이 증가함을 알 수 있도록 함

#### 환경

- VLC Media Player 활용
- RTSP 서버를 통해 음악 스트리밍 (rtsp://166.104.185.64:8088/pi)
- Attenuator 값(간섭하는 정도)을 60, 70, 80, 90으로 고정시킨 후, HARQ 값(4,3,2,1)에 따른 error rate 측정
- 음악 : '비루비 - 그리워하다'의 앞에 1분을 재생

### 실험 결과

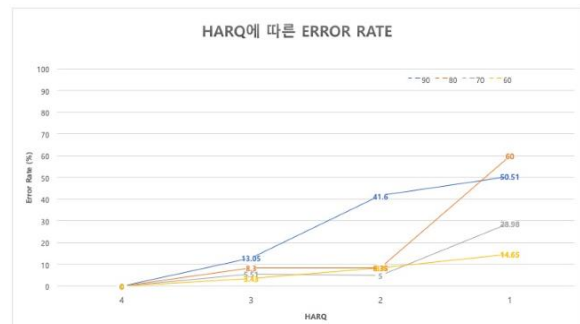
- HARQ값에 따른 음악이 끊기는 시간
- HARQ값이 낮을 수록 즉, 재전송 횟수가 적을수록 음악이 끊기는 시간이 긴 것을 알 수 있다.

HARQ ATT	1	2	3	4
90	00:00	0:07:83	0:25:21	0:30:31
80	00:00	0:05:00	0:05:01	0:36:34
70	00:00	0:03:31	0:03:00	0:17:39
60	00:00	0:02:06	0:05:04	0:08:79

### 실험 결과

- HARQ값에 따른 음악이 끊기는 시간에 따른 error rate

HARQ ATT	4	3	2	1
90	0	13.05	41.6	50.51
80	0	8.3	8.35	60
70	0	5.51	5	28.98
60	0	3.43	8.4	14.65



### 결론

- 실험 환경으로 인해 attenuator 값을 고정한 것과는 달리 RSRP 값이 지속적으로 변하여 attenuator가 80일 때 90보다 error rate이 큰 경우 발생하였다.
- 하지만, 재전송 횟수를 줄일 수록 error rate이 증가한다는 것을 확인할 수 있다.