

03. LTE

Synchronization

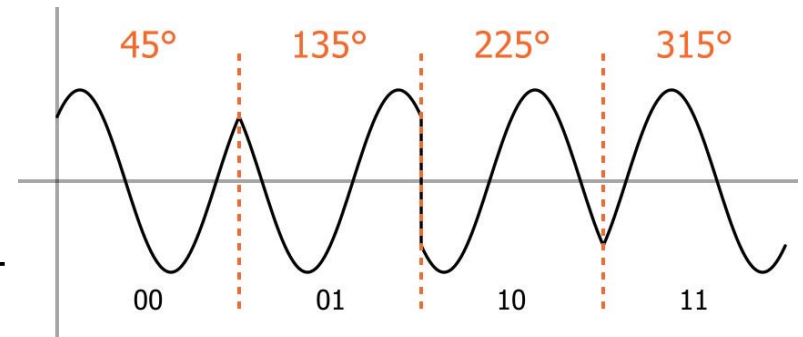
2019-1 Synchronization Study
Bit Processing System Lab.

Objectives

- synchronization 관련 기본 개념 이해
- LTE sync.에 대해 이해
 - Sync. signal 구조
 - Physical Cell ID (PCI)
 - 전체 sync. 과정

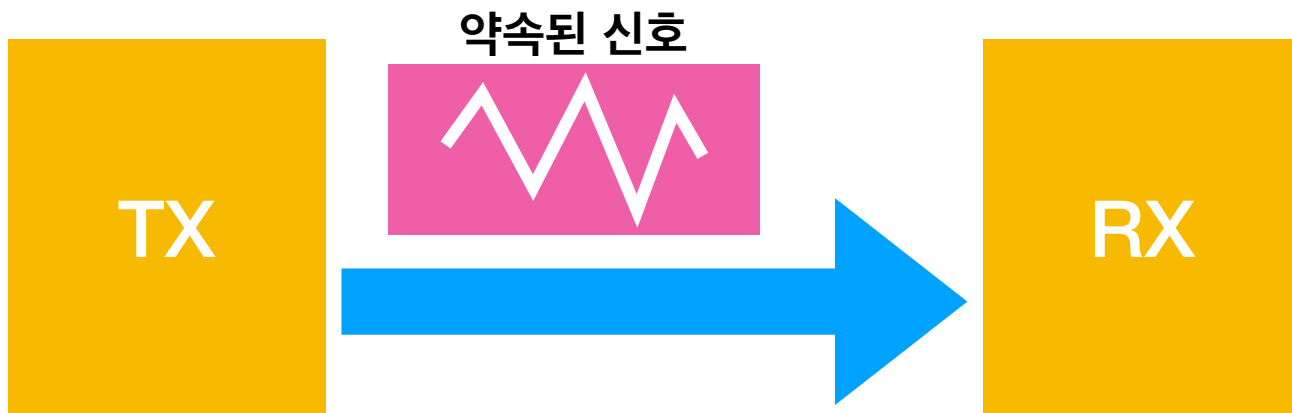
Synchronization 개념

- Synchronization이란
 - 통신 신호의 시작 timing을 파악하는 과정
 - 오늘날 디지털 통신에서 이루어지는 초기 필수 과정
- 시작 timing을 왜 파악해야 하는가?
 - 이를 알아야 demodulation을 할 수 있음
- 송신은 시작점을 알지만, 수신은 추정을 해야 할 대상임
 - multipath등 channel에 의한 delay



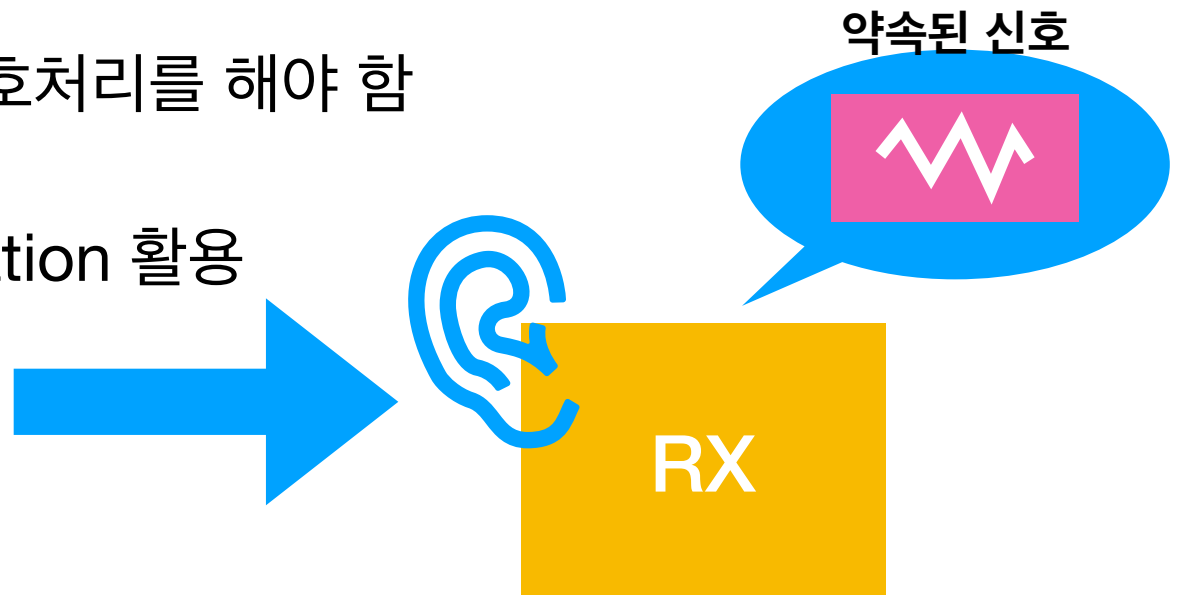
Synchronization 과정

- 어떻게 synchronization을 하는가?
 - 송/수신 양측에 서로 약속된 기저대역 신호 (preamble)를 사용
 - 복소수로 표현 가능한 symbol
 - 약속된 pattern/시점에 전송



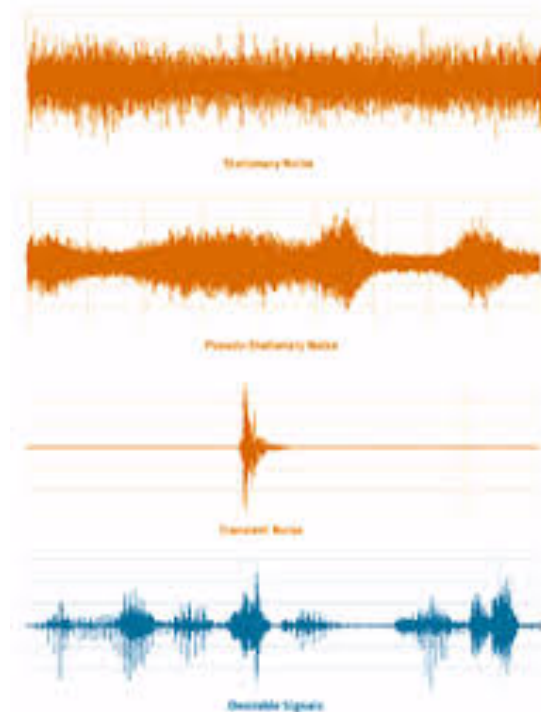
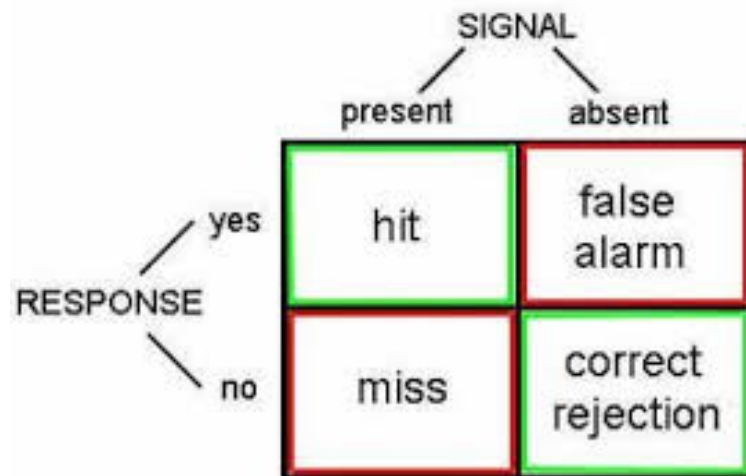
Synchronization 과정

- 송신 측에서는 약속된 대로 preamble을 보내면됨
- 수신 측에서는
 - preamble 신호를 찾아서 그 시작 timing을 파악함
 - **Detection**이라는 신호처리를 해야 함
 - 기본적으로 correlation 활용



Signal Detection

- 이것저것 섞인 수신 signal로부터 내가 원하는 패턴을 찾음
 - 디지털 통신의 경우 sampling된 discrete signal을 대상으로 함
- 궁극적으로 원래 symbol 수열에서 특정 pattern의 symbol 수열을 찾는 것
 - 복소수 수열의 상관관계를 따져야 함



Signal Detection

- Correlation이라는 통계적 기법 사용
 - 원래 패턴과 대상 signal에 대한 상관도를 숫자로 표현

$$r_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

- x, y가 비슷하면 :
==> r값이 커짐
- x,y값이 서로 다르면 :
==> r값이 작아짐

Signal Detection

- 일반적으로 preamble은 mean=0, power = 1임
- 따라서 식은 더 간략화가 될 수 있음

$$r_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

The diagram illustrates simplifications in the correlation coefficient formula. Red circles highlight the mean terms \bar{x} and \bar{y} in the numerator, with an arrow pointing to $=0$. Red brackets highlight the variance terms $\sum (x_i - \bar{x})^2$ and $\sum (y_i - \bar{y})^2$ in the denominator, with an arrow pointing to $=1$.

Signal Detection

- 하지만 우리가 대상으로 하는 패턴은 symbol이며, 복소수 수열임
- 복소수에 대한 상관관계 : complex inner product 사용

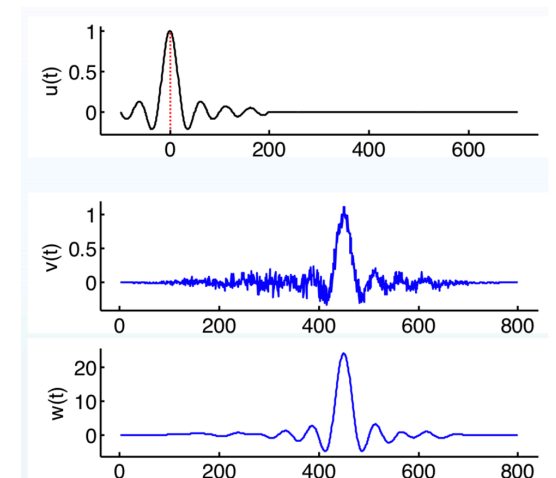
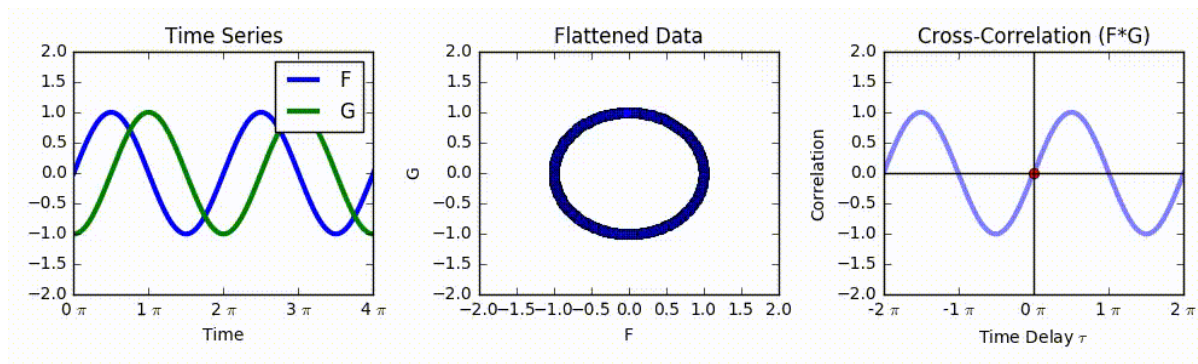
$$metric = \mathbf{a}^* \cdot \mathbf{b}$$

- metric값이 높을수록 a와 b가 닮았다는 뜻임

Cross-Correlation

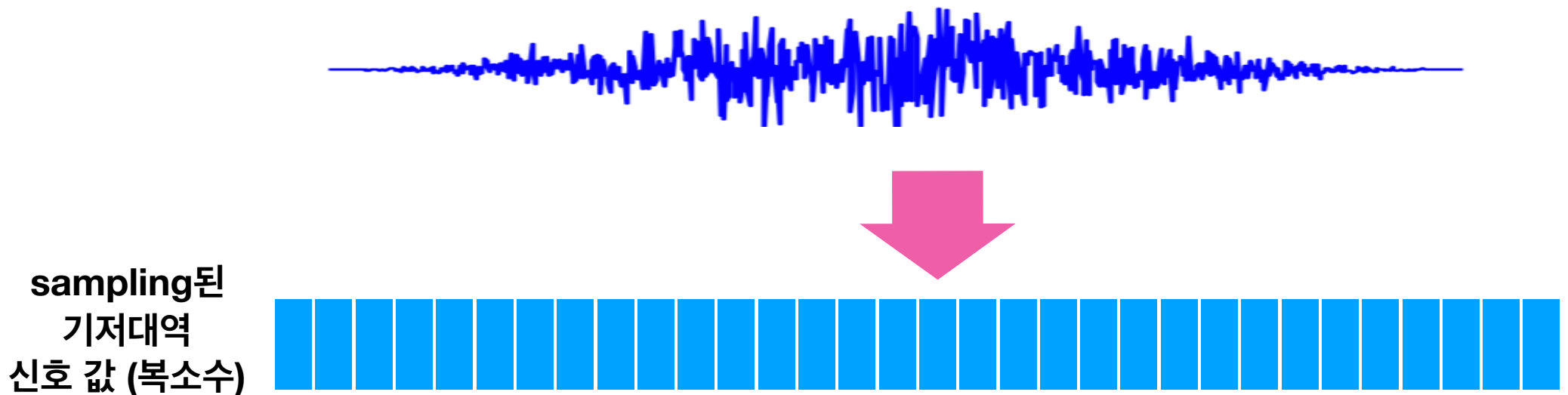
- 신호 처리 관점에서의 correlation
 - 신호는 n 이라는 시간에 따라 변하는 값임
 - n 에 따른 상관도의 변화 양상을 봐야 함
- Cross-correlation : complex inner product를 시간 n 에 따라
 쪽 해봄

$$(f \star g)[n] \triangleq \sum_{m=-\infty}^{\infty} \overline{f[m]} g[m+n]$$



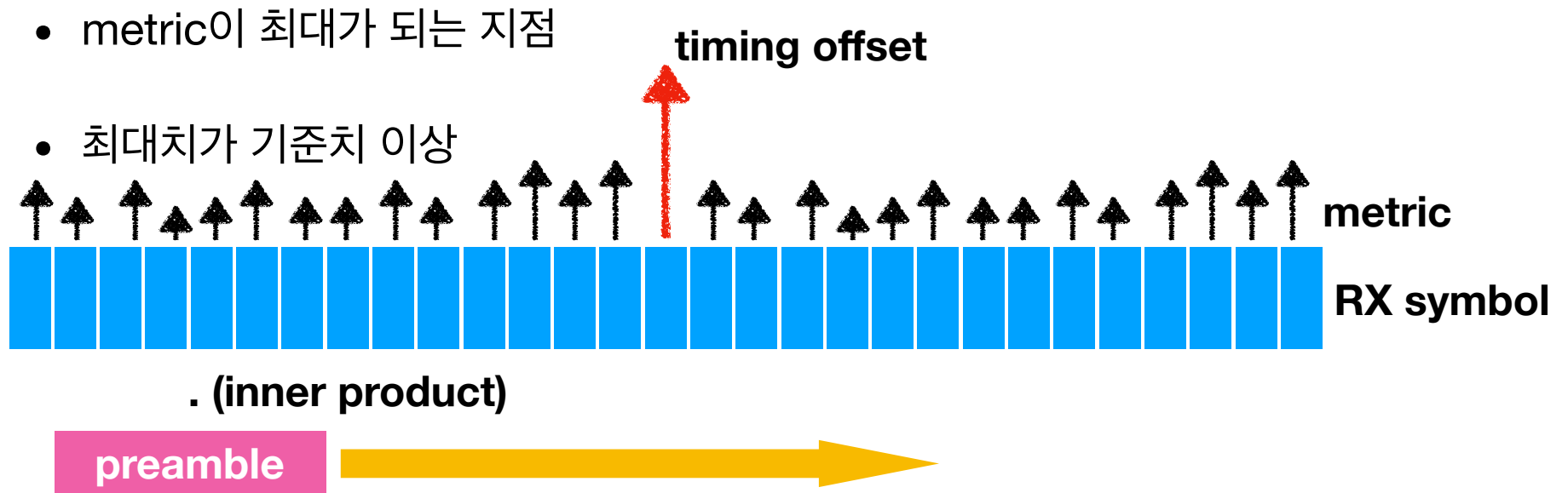
Synchronization Signal Detection 과정

- Oversampling
 - timing 계산을 위해 원래 Nyquist theorem에 의한 sampling rate보다 더 자주 sampling을 함
 - sampling된 수신 symbol들을 저장



Synchronization Signal Detection 과정

- cross-correlation 계산
 - signal demodulation (I/Q값 추출)
 - timing별로 shift를 하면서 inner product 계산
- metric에 따라 timing 결정



Zhadoff-Chu Sequence

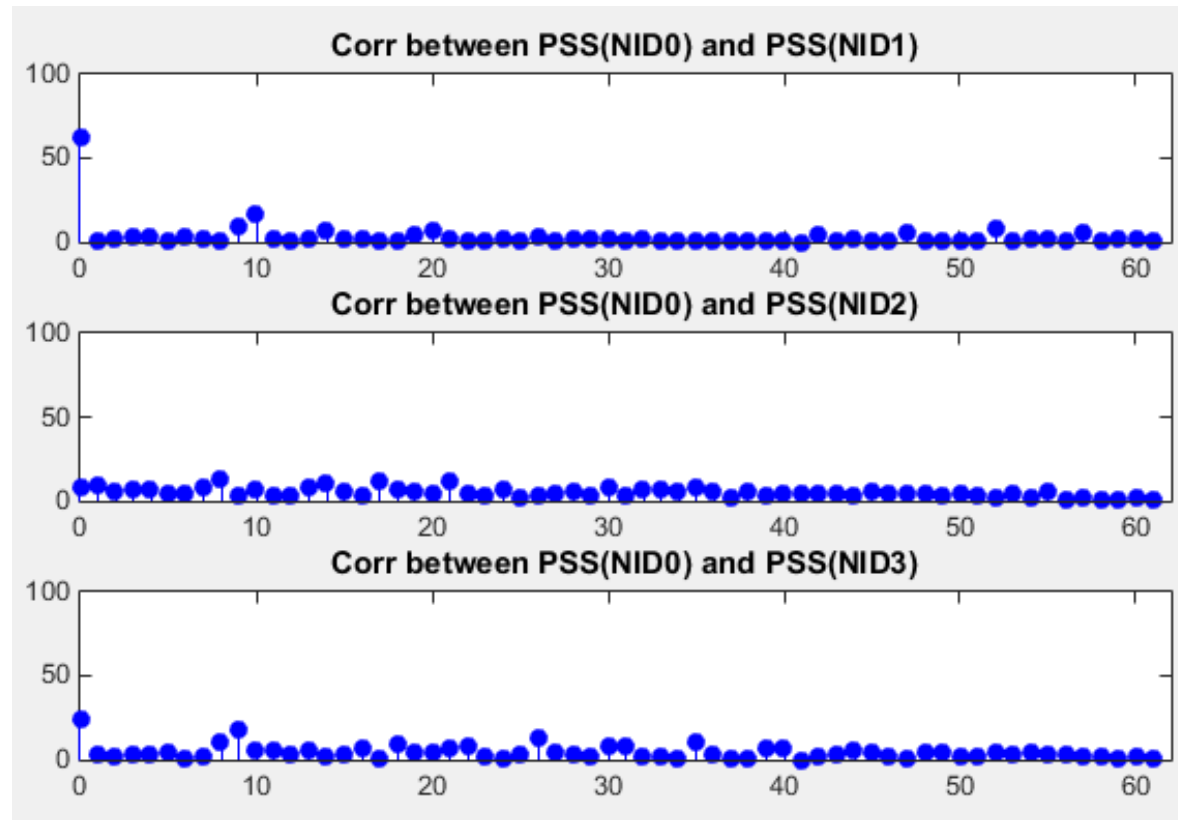
- LTE에서 preamble로 쓰이는 특정 signal

$$x_u(n) = \exp \left(-j \frac{\pi u n (n + c_f + 2q)}{N_{ZC}} \right) \quad c_f = N_{ZC}$$

- u 값에 따라 수열 패턴이 달라짐
- 특성
 - N_{ZC} 가 주기
 - N_{ZC} 가 prime이면 DFT 결과는 또다른 Zhadoff-Chu sequence의 conjugated/scaled/time scaled 버전임
 - Auto-correlation(자신에 대한 cross-correlation) 특성이 좋음
 - 서로 다른 ZC seq.에 대한 cross correlation 특성이 좋음

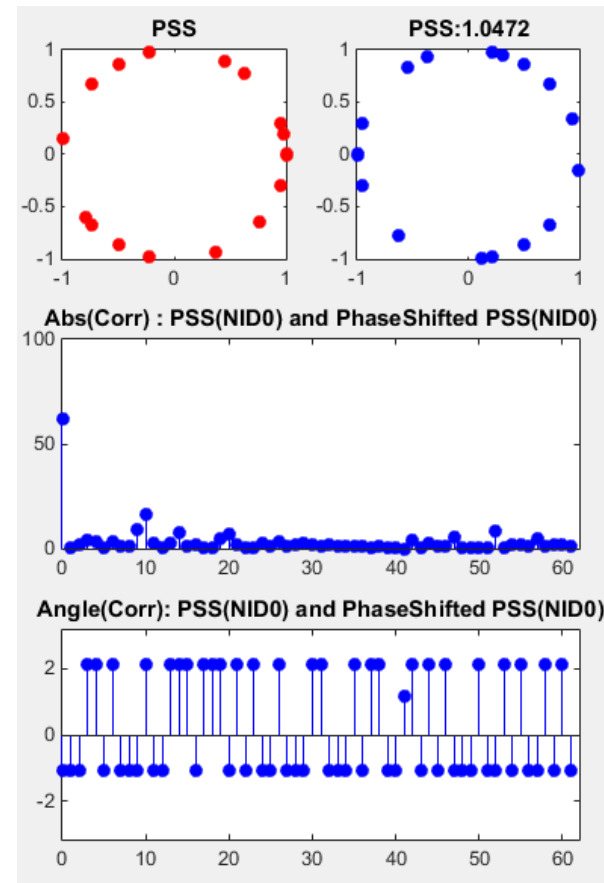
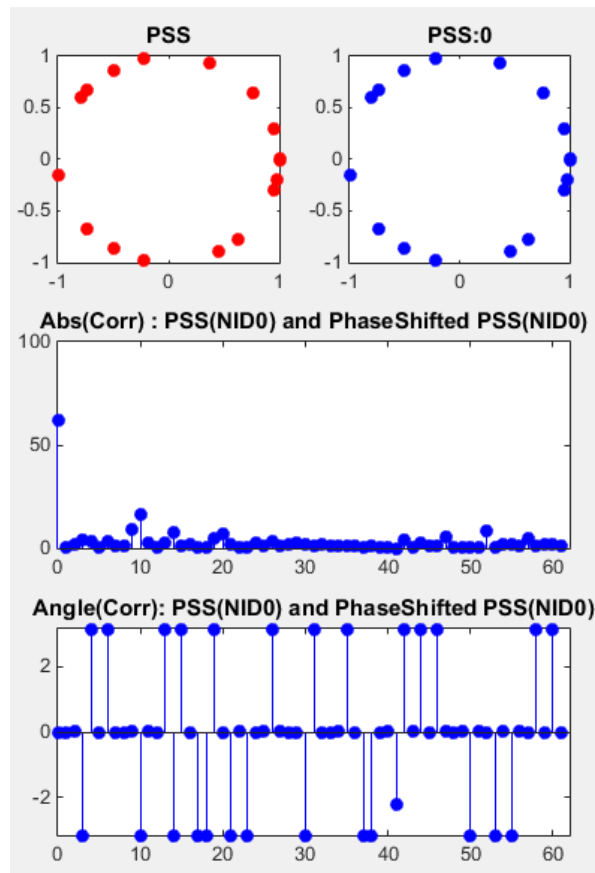
Zhadoff-Chu Sequence

- 예시 : 서로 다른 PSS간 cross-correlation이 매우 작음



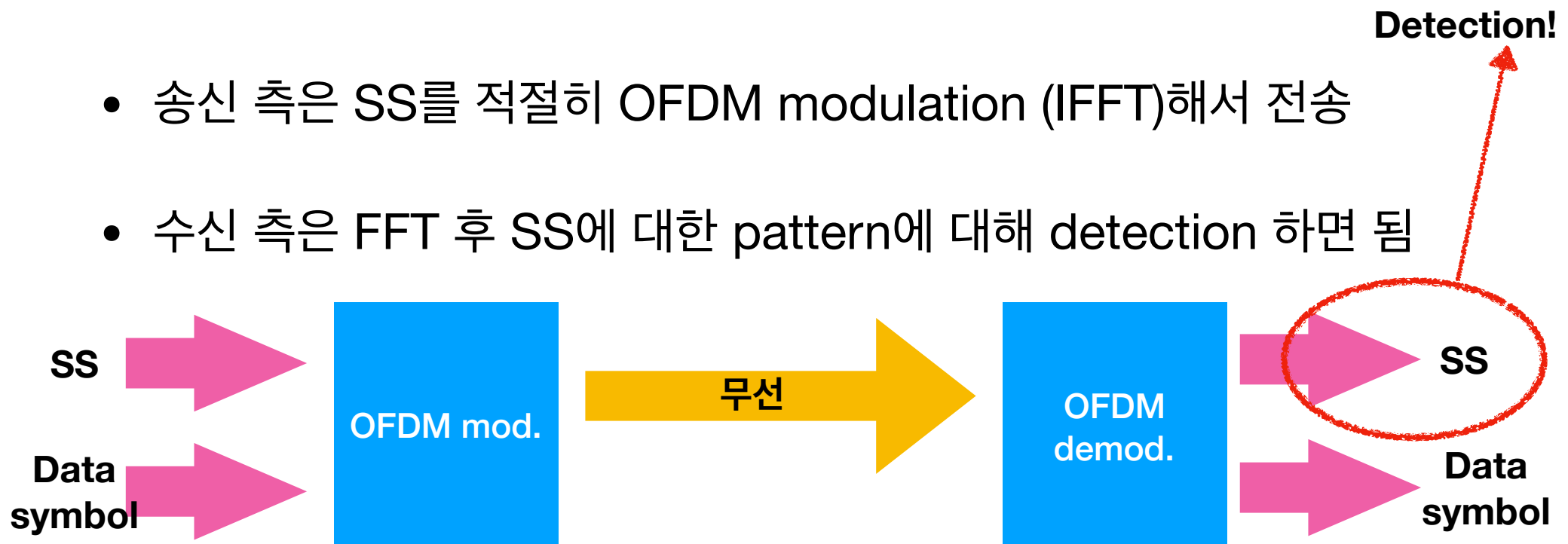
Zhadoff-Chu Sequence

- 예시 : Phase shift에도 강인함
- phase rotation에 대한 추정도 어느정도 가능



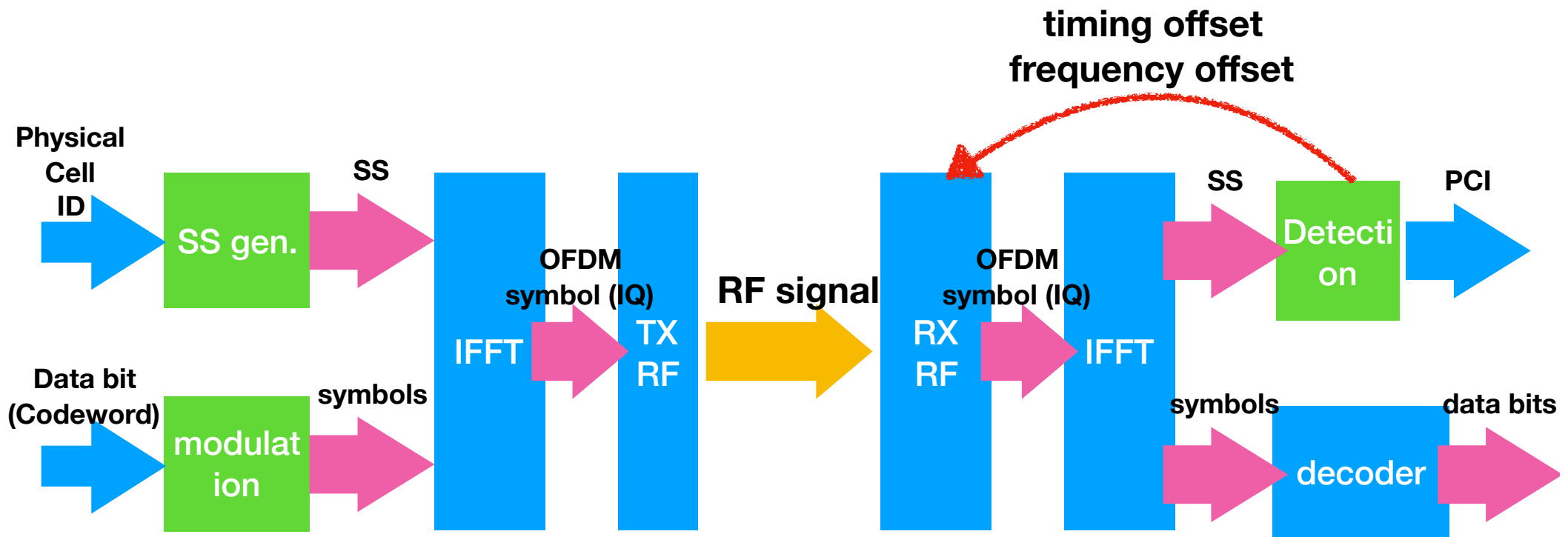
LTE Synchronization : 개념

- LTE에서 정의한 synchronization signal (preamble, SS)을 찾음
 - OFDM modulation 이전 단계에서의 symbol level의 약속된 신호
- 송신 측은 SS를 적절히 OFDM modulation (IFFT)해서 전송
- 수신 측은 FFT 후 SS에 대한 pattern에 대해 detection 하면 됨



LTE Synchronization : 과정

- 전체 과정



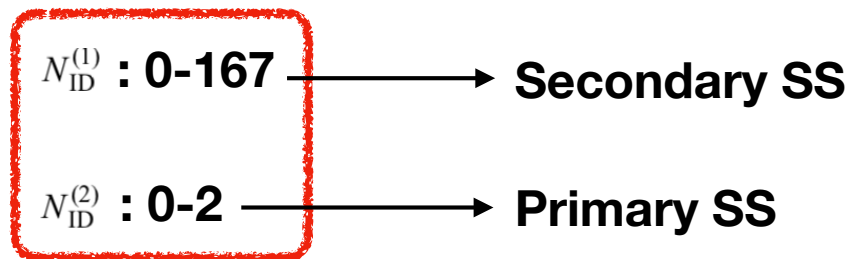
Physical Signal

- transport channel layer에서 내려오지 않고, physical channel level에서 자체적으로 만들어내는 signal
 - modulation 과정을 거치지 않고 바로 complex signal 형태로 생성
 - synchronization signal 혹은 reference signal
- Synchronization Signal (SS) : 동기화를 위한 signal
 - Primary SS (PSS) : 1차적으로 timing을 잡기 위한 signal
 - Secondary SS (SSS) : 2차적으로 PCI를 알아내고 timing에 대한 확신을 하기 위한 signal

Physical Cell ID (PCI)

- 물리적인 기지국 번호
 - 총 504개 (0~503)
 - 고유 식별번호는 아니며, local한 식별은 됨
- Synchronization signal을 통해 파악 가능
- NID(1), NID(2)로 구성

$$N_{\text{ID}}^{\text{cell}} = 3N_{\text{ID}}^{(1)} + N_{\text{ID}}^{(2)}$$



PSS and SSS

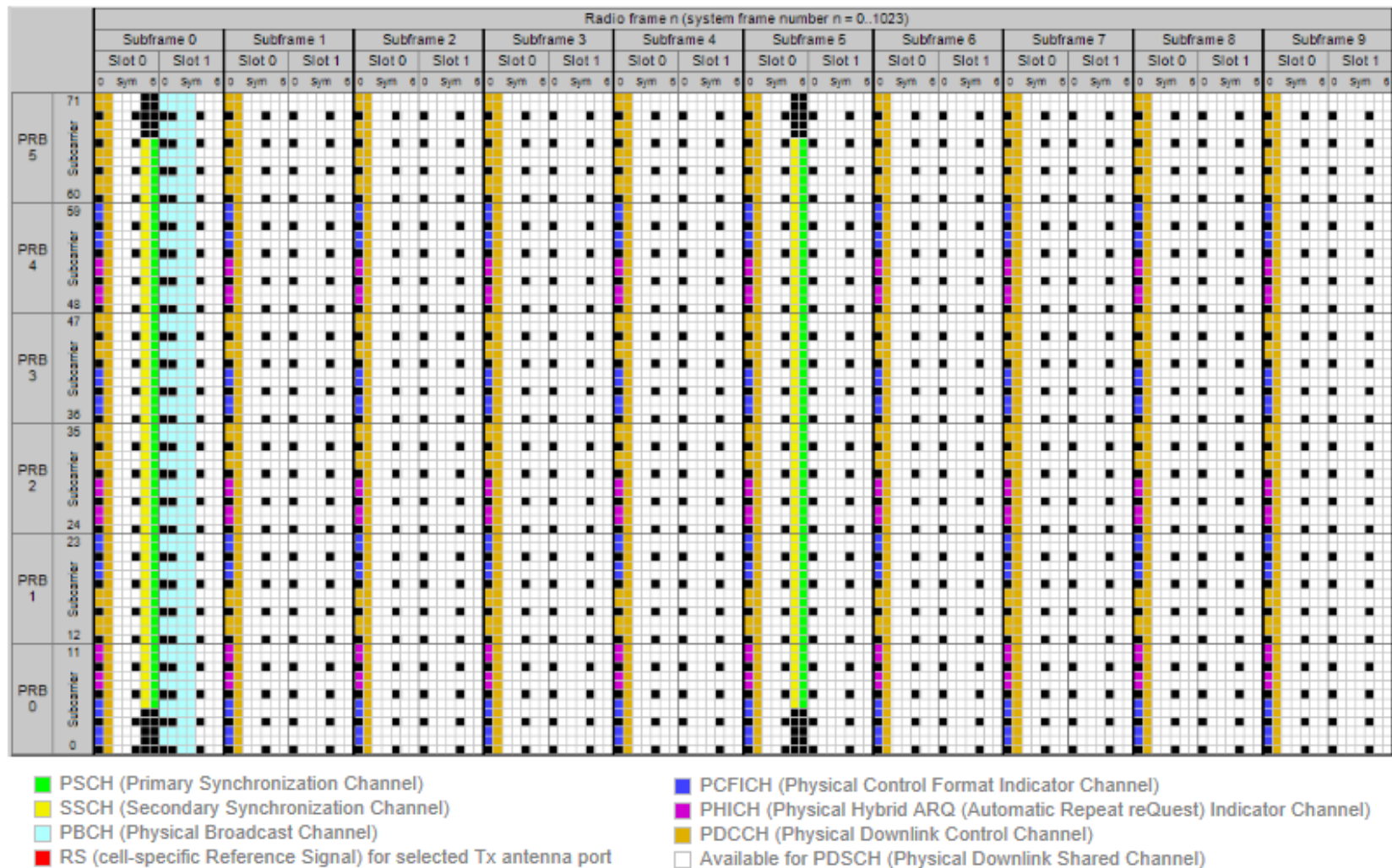
- PSS/SSS 구조
 - 각각 총 62개의 symbol로 이루어짐
 - Resource grid상에서의 위치
 - radio frame (10ms)마다 2군데에 들어감
 - time : subframe 0/5, slot 0, 10, OFDM symbol 6, 7
 - PSS는 동일 pattern, SSS는 다른 pattern이 들어감
 - freq. : 가운데 지점

$$a_{k,l} = d(n), \quad n = 0, \dots, 61$$

$$k = n - 31 + \frac{N_{\text{RB}}^{\text{DL}} N_{\text{sc}}^{\text{RB}}}{2}$$

PSS and SSS

- PSS/SSS 구조



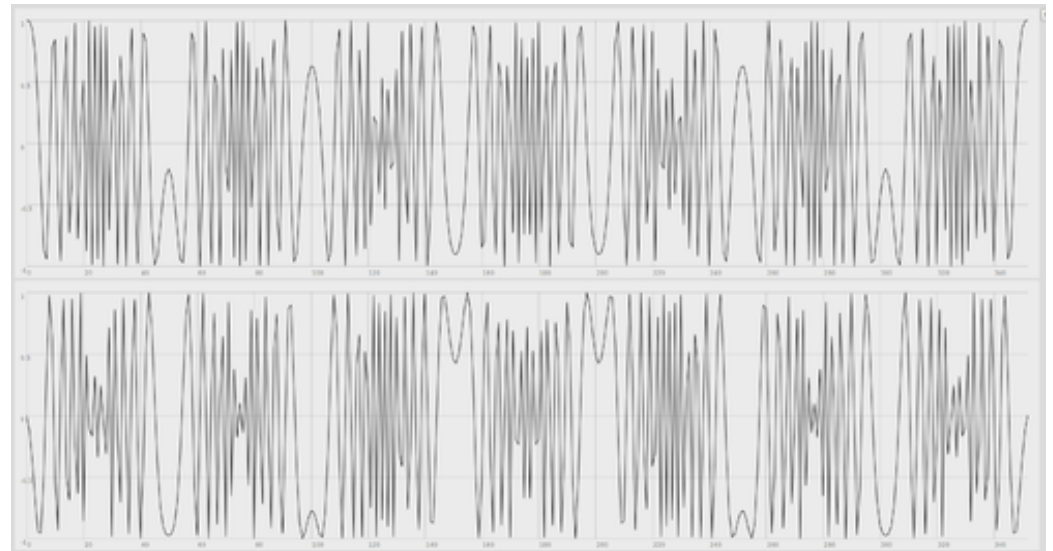
PSS Generation

- Frequency-domain Zadoff-Chu sequence

$$d_u(n) = \begin{cases} e^{-j\frac{\pi un(n+1)}{63}} & n = 0,1,\dots,30 \\ e^{-j\frac{\pi u(n+1)(n+2)}{63}} & n = 31,32,\dots,61 \end{cases}$$

- root sequence index u

| $N_{ID}^{(2)}$ | Root index u |
|----------------|----------------|
| 0 | 25 |
| 1 | 29 |
| 2 | 34 |



PSS Generation

- PSS는 신호의 존재를 파악하는데 적합
 - Auto-correlation을 통해 detection이 용이
 - 신호가 전송되는 시작점을 판단할 수 있어 timing offset 추정
이 비교적 용이함
 - 다양한 무선채널 현상이 일어나도 강인함
- phase rotation 추정 가능
 - phase에 대한 보상 가능

SSS Generation

- Interleaved concatenation of two length-31 binary sequences
 - scrambling이 됨 (scrambling sequence는 PSS로부터 유도)
 - subframe 0, 5번 내 SSS가 서로 다른 sequence가 들어감

$$d(2n) = \begin{cases} s_0^{(m_0)}(n)c_0(n) & \text{in subframe 0} \\ s_1^{(m_1)}(n)c_0(n) & \text{in subframe 5} \end{cases}$$

$$d(2n+1) = \begin{cases} s_1^{(m_1)}(n)c_1(n)z_1^{(m_0)}(n) & \text{in subframe 0} \\ s_0^{(m_0)}(n)c_1(n)z_1^{(m_1)}(n) & \text{in subframe 5} \end{cases}$$

$$0 \leq n \leq 30$$

$$m_0 = m' \bmod 31$$

$$m_1 = (m_0 + \lfloor m'/31 \rfloor + 1) \bmod 31$$

$$m' = N_{\text{ID}}^{(1)} + q(q+1)/2, \quad q = \left\lfloor \frac{N_{\text{ID}}^{(1)} + q'(q'+1)/2}{30} \right\rfloor, \quad q' = \lfloor N_{\text{ID}}^{(1)} / 30 \rfloor$$

SSS Generation

- $s(n)$: 특정 sequence의 두가지 cyclic shift 버전

$$s_0^{(m_0)}(n) = \tilde{s}((n + m_0) \bmod 31)$$

$$s_1^{(m_1)}(n) = \tilde{s}((n + m_1) \bmod 31)$$

where $\tilde{s}(i) = 1 - 2x(i)$, $0 \leq i \leq 30$, is defined by

$$x(\bar{i} + 5) = (x(\bar{i} + 2) + x(\bar{i})) \bmod 2, \quad 0 \leq \bar{i} \leq 25$$

with initial conditions $x(0) = 0$, $x(1) = 0$, $x(2) = 0$, $x(3) = 0$, $x(4) = 1$.

SSS Generation

- $c(n)$: 두가지 scrambling sequence가 존재하며 PSS detection 결과값에 따라 다르게 생성

$$c_0(n) = \tilde{c}((n + N_{\text{ID}}^{(2)}) \bmod 31)$$

$$c_1(n) = \tilde{c}((n + N_{\text{ID}}^{(2)} + 3) \bmod 31)$$

$\tilde{c}(i) = 1 - 2x(i)$, $0 \leq i \leq 30$, is defined by

$$x(\bar{i} + 5) = (x(\bar{i} + 3) + x(\bar{i})) \bmod 2, \quad 0 \leq \bar{i} \leq 25$$

with initial conditions $x(0) = 0$, $x(1) = 0$, $x(2) = 0$, $x(3) = 0$, $x(4) = 1$.

SSS Generation

- $z(n)$: 두가지 scrambling sequence는 특정 sequence를 cyclic shift함

$$z_1^{(m_0)}(n) = \tilde{z}((n + (m_0 \bmod 8)) \bmod 31)$$

$$z_1^{(m_1)}(n) = \tilde{z}((n + (m_1 \bmod 8)) \bmod 31)$$

where m_0 and m_1 are obtained from Table 6.11.2.1-1 and $\tilde{z}(i) = 1 - 2x(i)$, $0 \leq i \leq 30$, is defined by

$$x(\bar{i} + 5) = (x(\bar{i} + 4) + x(\bar{i} + 2) + x(\bar{i} + 1) + x(\bar{i})) \bmod 2, \quad 0 \leq \bar{i} \leq 25$$

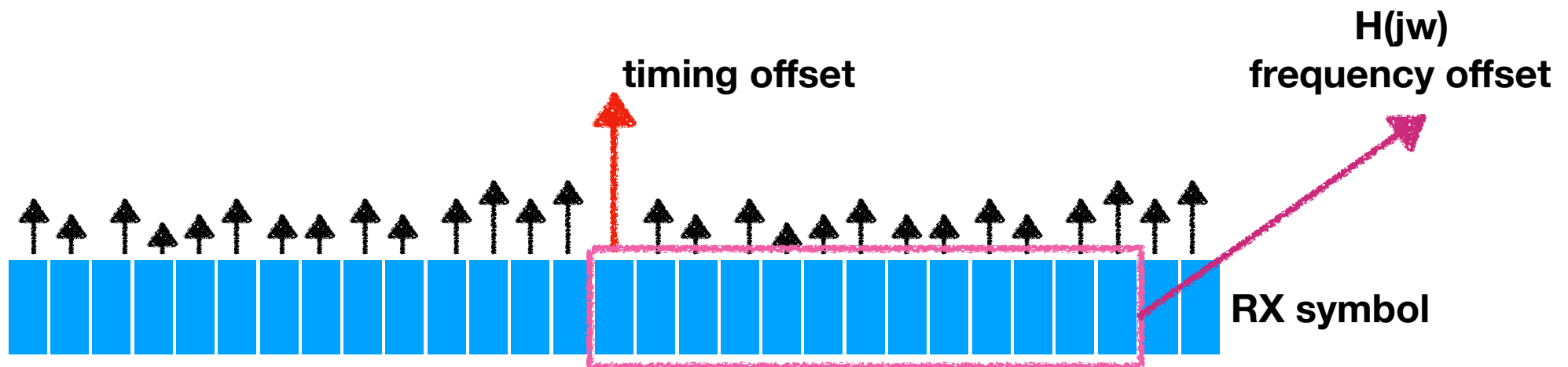
with initial conditions $x(0) = 0$, $x(1) = 0$, $x(2) = 0$, $x(3) = 0$, $x(4) = 1$.

PSS / SSS Detection

- 전체 과정
 - PSS detection : 있는 그대로의 time-domain signal에 대해 추정
 - timing offset (slot boundary) 추정
 - SSS에 대한 channel 보정 관련 정보 추정 (frequency offset, channel estimation)
 - SSS detection : 일부 보정 후 세밀하게 추정
 - NID (PCI) 추정
 - frame boundary 추정
 - PBCH decoding : timing을 알면 기본적으로 수신 처리가 가능하므로, broadcasting되는 정보에 대해 시범적으로 decoding
 - timing 정보 등에 대한 확인사살

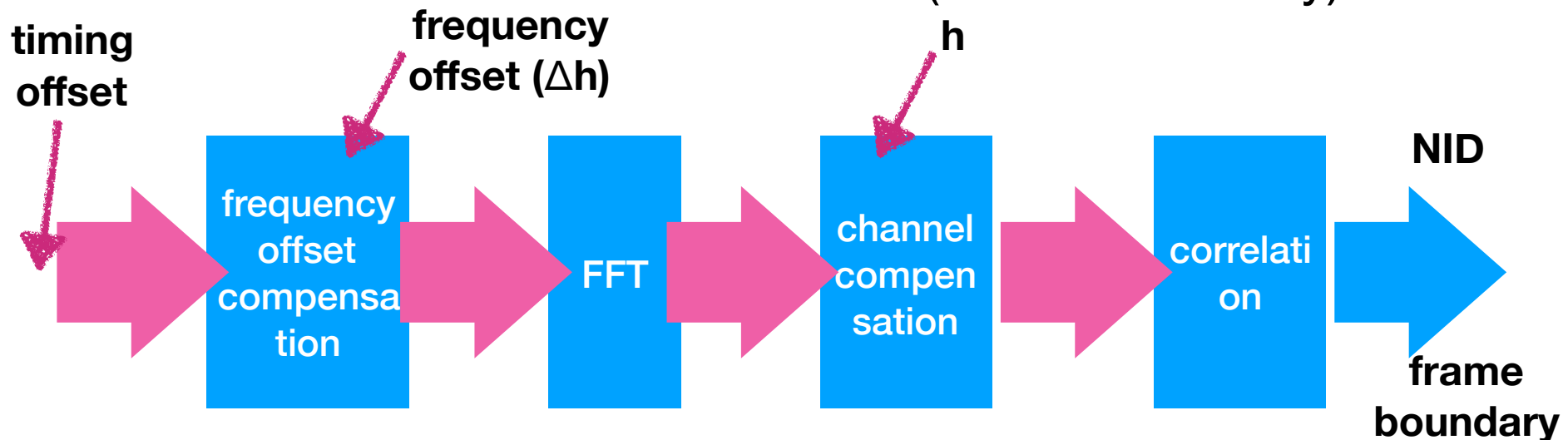
PSS Detection

- Time domain에서의 cross-correlation 수행
 - 3가지 Zadoff-Chu sequence에 대해 여러가지 timing에 대해서 metric을 구함
 - metric이 threshold이상이고 최대가 되는 지점에서의 timing 값을 구함
- PSS는 correlation 특성이 좋은 sequence를 사용하므로, time-domain에서 보정 없이 detection이 가능



SSS Detection

- Frequency domain correlation 수행
 - PSS로 추정된 timing offset, frequency offset, H 정보 활용
 - 충분히 보정 후 168개 후보 패턴으로부터 가장 적합한 것 찾기
- subframe 0/5위치를 찾을 수 있음 (frame boundary)



실습

- MATLAB을 이용해서 Synchronization signal detection에 대해 실습
 - sync_sim.m #5을 base code로 사용
 - gen_PSS, gen_SSS 함수 사용
 - PSS detection에 대한 code 구현
 - SSS detection에 대한 code 구현

실습

- PSS detection
 - 반복을 통해서 correlation을 계산해서 가장 적합한 NID2 값 및 timing offset 계산
 - PSS 신호 생성
 - 수신된 신호 rx_demod의 각 timing에 대해서 correlation 계산
 - correlation값이 최대가 되는 NID 및 timing (slot boundary)을 찾음

실습

- SSS detection
 - 가장 유사한 SSS 신호를 찾아서 NID1를 추정
 - 수신 신호 (rx_demod) 로부터 SSS 관련 symbol 추출
 - 각 NID1 후보 값에 따라 SSS 신호를 생성하고, correlation 계산
 - subframe 0/5 여부를 고려해야 함
 - 가장 correlation이 높은 NID1 값에 대해 이전에 구한 NID2와 조합하여 PCI 계산
 - 이때 frame0이 시작되는 시작점 timing을 추정

다음주 연습

- 다음주에 진행할 채널 추정 및 PBCH decoding 과정에 대한 연습 진행
 - 채널 추정에 대한 기본 개념 조사
 - pilot 기본 개념
 - 채널 추정에 대한 일반적인 방법
 - LTE의 reference signal에 대한 개념 및 정의
 - PBCH decoder에 대한 기본 개념 조사
 - PBCH에 대한 기본 개념 조사
 - LTE에서의 scrambling, channel coding, rate matching, CRC 등 처리에 대한 기본 개념 조사