The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2025-6

Satellite Navigation Science and Technology for Africa

23 March - 9 April, 2009

Introduction to GPS Receiver Design Principles (Part 3)

Ward Phillip. W.
Navward
U.S.A.

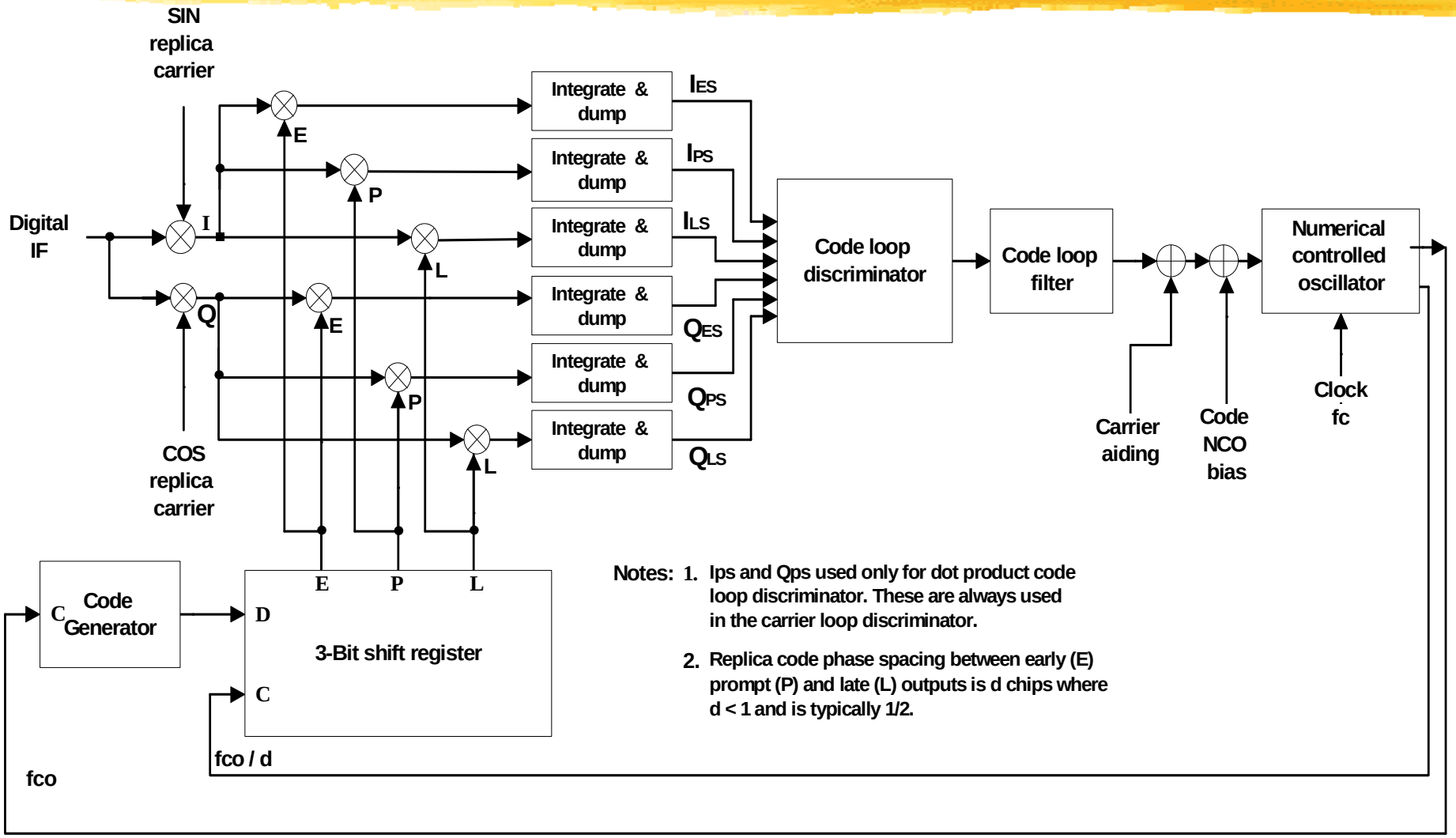
Session III
Code tracking loop design

- Generic GPS receiver code tracking loop block diagram
- Approximation techniques for computing I and Q signal envelopes
- Common delay lock loop (DLL) discriminators
- Comparison of DLL discriminators S-curves
- Code correlation process for three different replica code phases
- Correlation for replica codes: 1/2 chip early, 1/4 chip early, aligned and 1/4 chip late
- Code discriminator output versus replica code offset

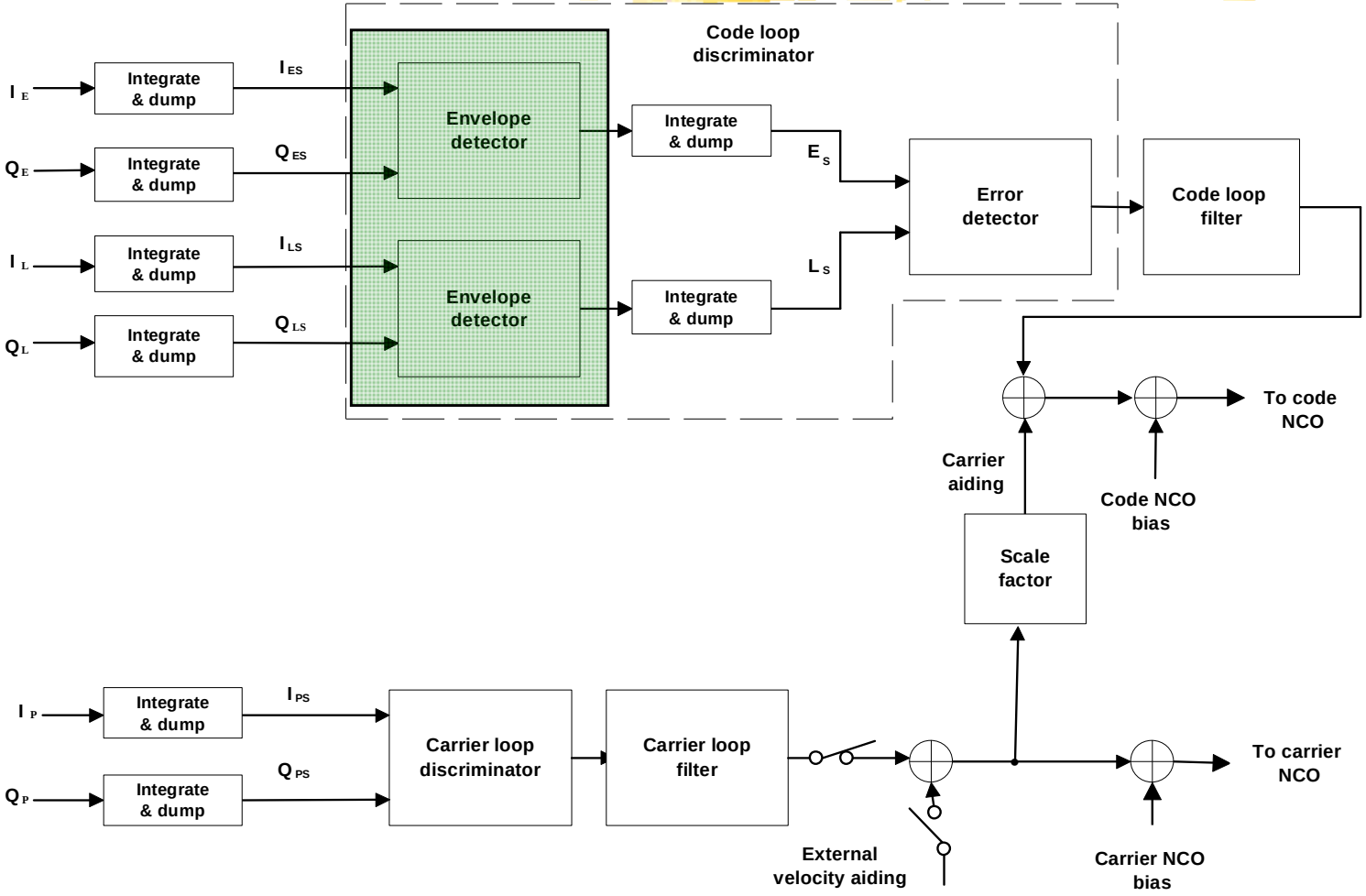
Generic GPS receiver code tracking loop block diagram

- Carrier tracking loop description spread out over several block diagrams
  - Replica code generation hardware
  - Carrier then code wipe-off hardware
  - Predetection integration hardware
  - Baseband software
    - Code discriminator and filter
- Discriminator defines code loop type
  - Non-coherent or coherent

Generic GPS receiver code tracking loop block diagram



Approximation techniques for computing I and Q signal envelopes



Approximation techniques for computing I and Q signal envelopes

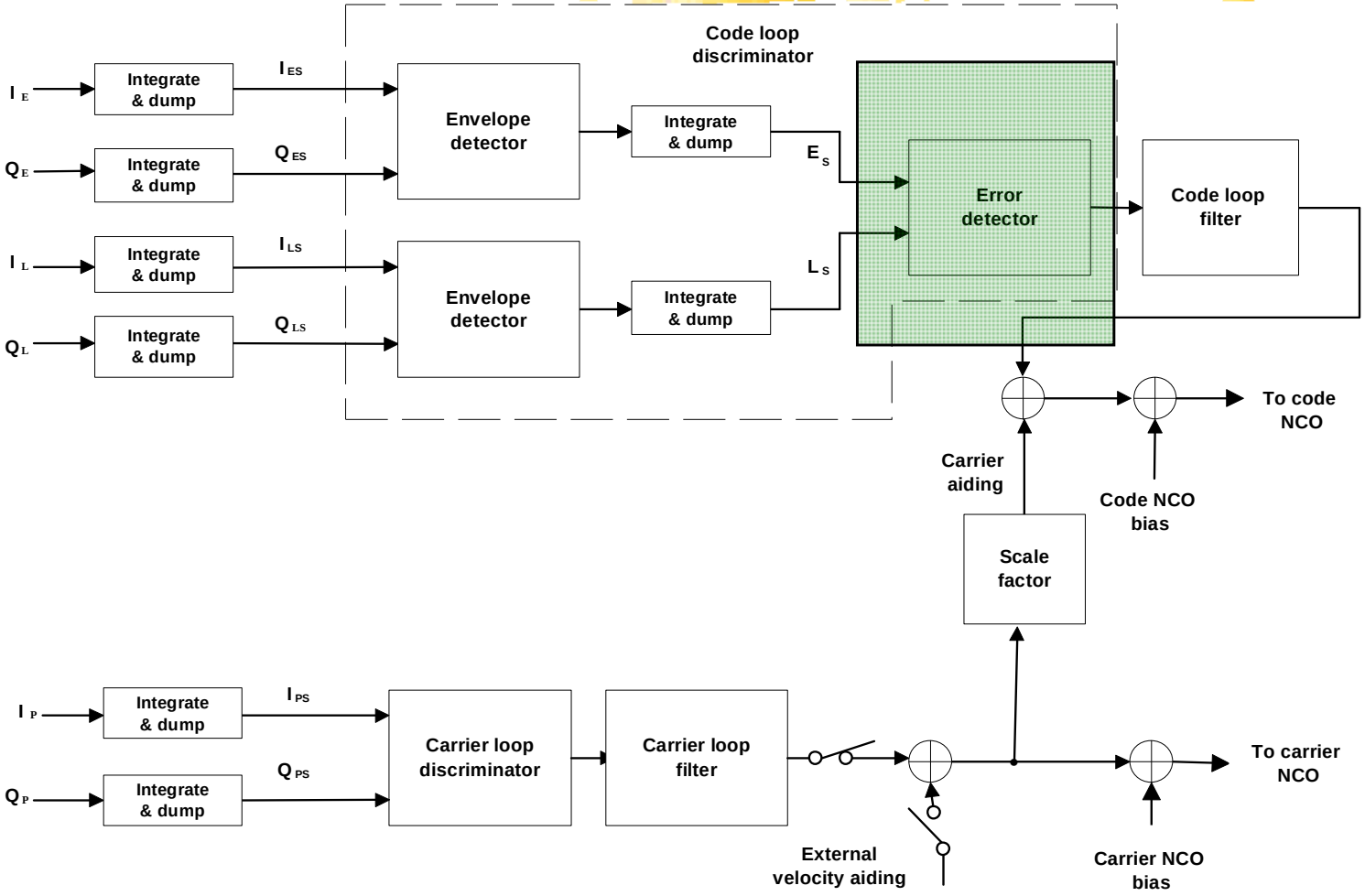
- JPL approximation of $A_{ENV} = \sqrt{I^2 + Q^2}$

(most accurate, used in track):

- A_{JPL} = X+1/8Y if X≥3Y
- A_{JPL}= 7/8X+1/2Y if X<3Y
- where: X = MAX(|I|,|Q|); Y = MIN(|I|,|Q|)

- Robertson approximation (used in search):
- A_{RBN} = MAX (|I|+1/2|Q|, |Q|+1/2|I|)

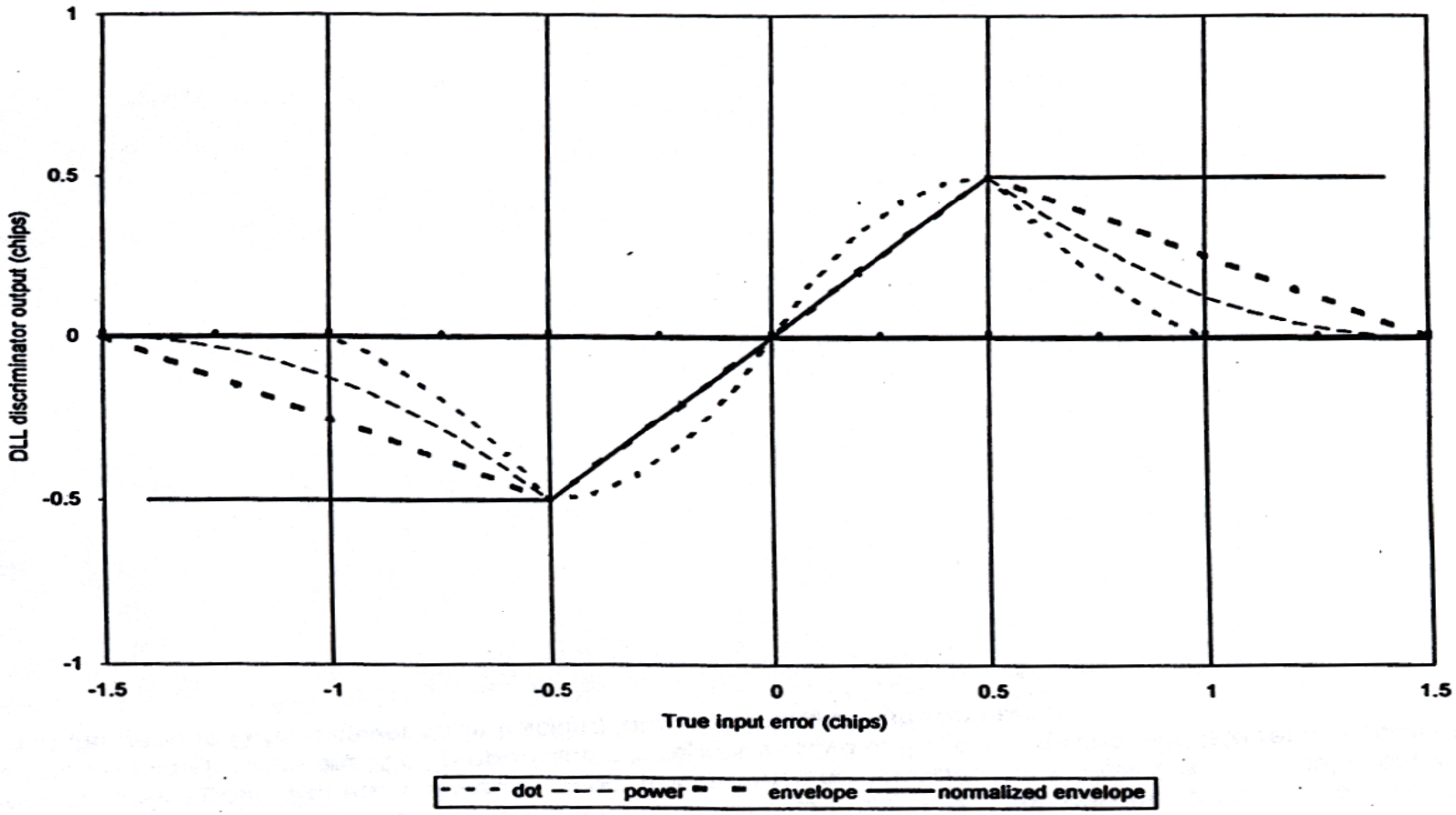
Common delay lock loop (DLL) discriminators



Common DLL discriminators

Discriminator algorithm	Characteristics
$(I_{ES} - I_{LS})\, I_{PS} + (Q_{ES} - Q_{LS})\, Q_{PS}$	Dot product power. This is the only DLL discriminator which uses all three correlators and this results in the lowest baseband computational load. For ½ chip correlator spacing, it produces nearly true error output within ± ½ chip of input error.
$\frac{1}{2}\left[(I_{ES}^2 + Q_{ES}^2) - (I_{LS}^2 + Q_{LS}^2)\right]$	Early minus late power. Moderate computational load. Essentially the same DLL discriminator error performance as early minus late envelope within ± ½ chip of input error.
$\frac{1}{2}\left[\sqrt{I_{ES}^2 + Q_{ES}^2} - \sqrt{I_{LS}^2 + Q_{LS}^2}\right]$	Early minus late envelope. Higher computational load. For ½ chip correlator spacing, produces good tracking error within ± ½ chip of input error.
$\frac{1}{2}\left[\frac{\sqrt{(I_{ES}^2 + Q_{ES}^2} - \sqrt{I_{LS}^2 + Q_{LS}^2}}{\sqrt{I_{ES}^2 + Q_{ES}^2} + \sqrt{I_{LS}^2 + Q_{LS}^2}}\right]$	Early minus late envelope normalized by the early plus late envelope to remove amplitude sensitivity. Highest computational load. For ½ chip correlator spacing, produces good tracking error within less than ± 1.5 chip of input error. Becomes unstable (divide by zero) at ± 1.5 chip input error.

Comparison of DLL discriminators S-curves

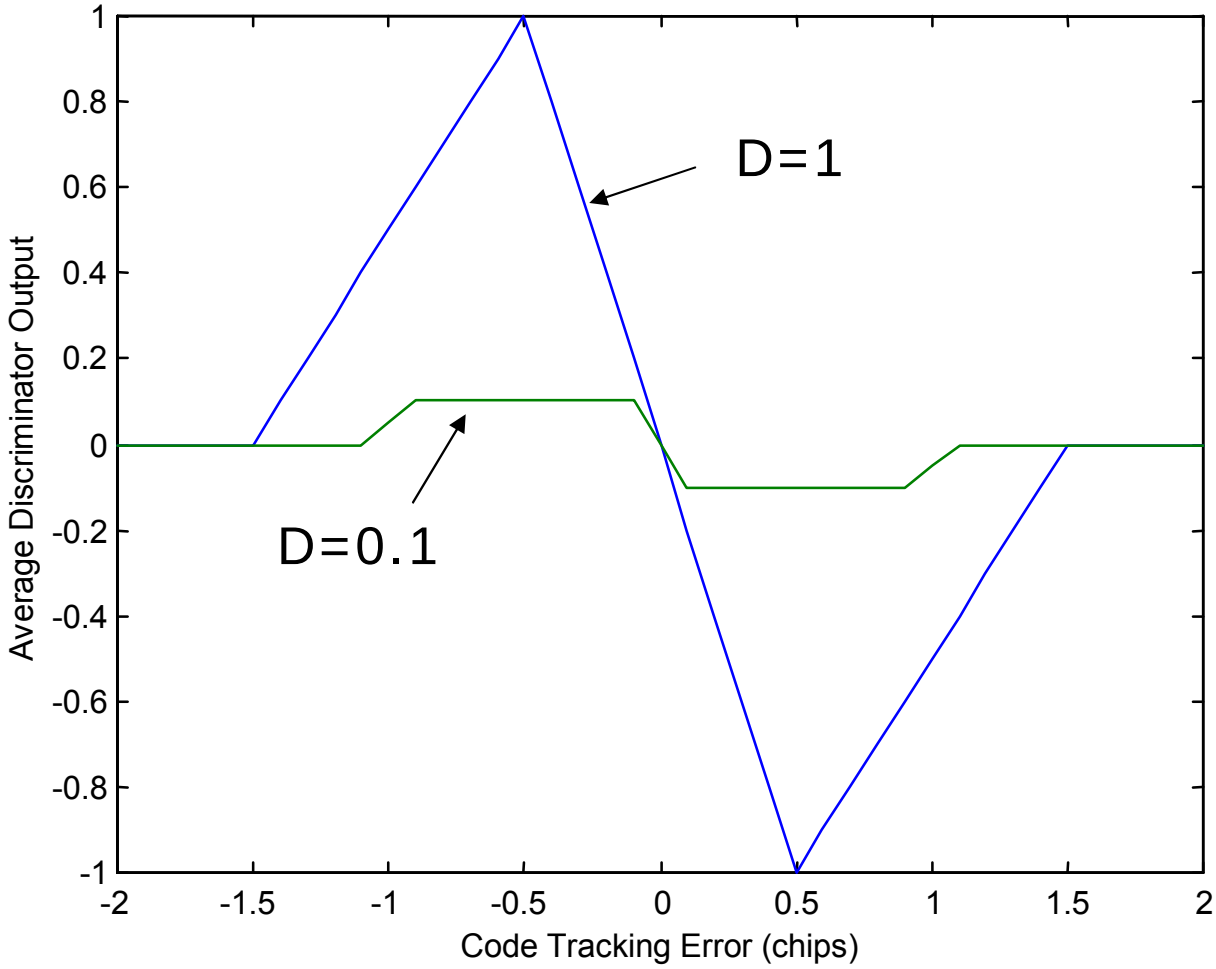


Coherent DLL discriminator

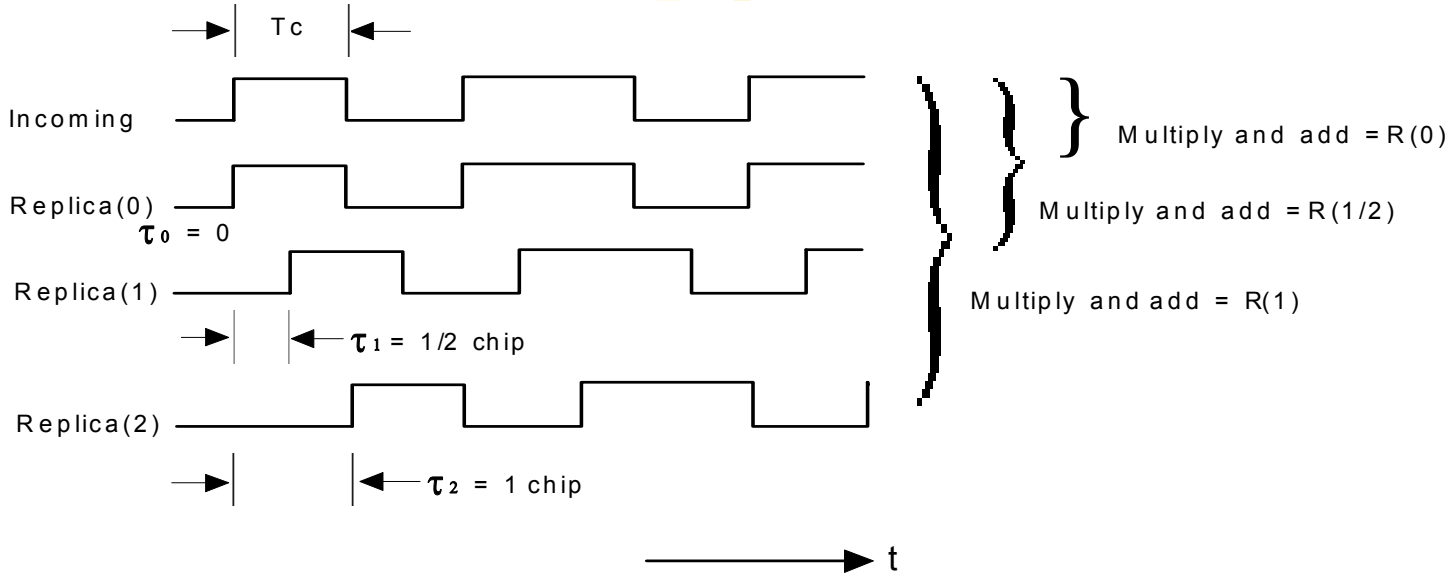
- Carrier tracking loop must be in phase lock to support coherent DLL operation
- In this case all Q_{ES} signals are nearly zero, so only the I_{ES} signals are used
- This is the most precise (lowest noise) but also the most fragile DLL tracking mode
- Coherent DLL discriminator algorithm:

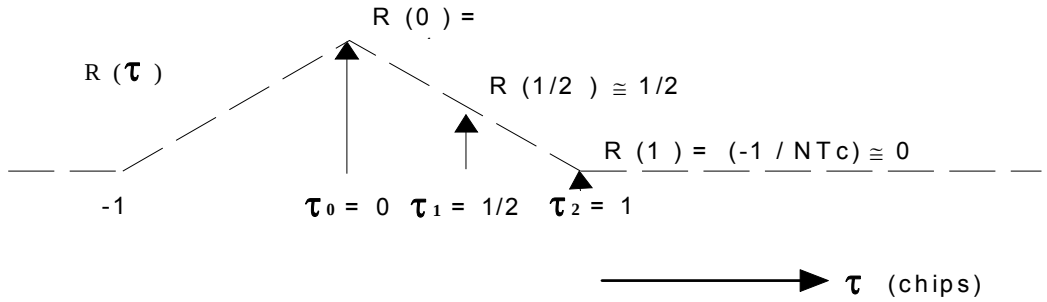
$$(I_{ES} - I_{LS})(I_{PS})$$

Coherent DLL S-curve

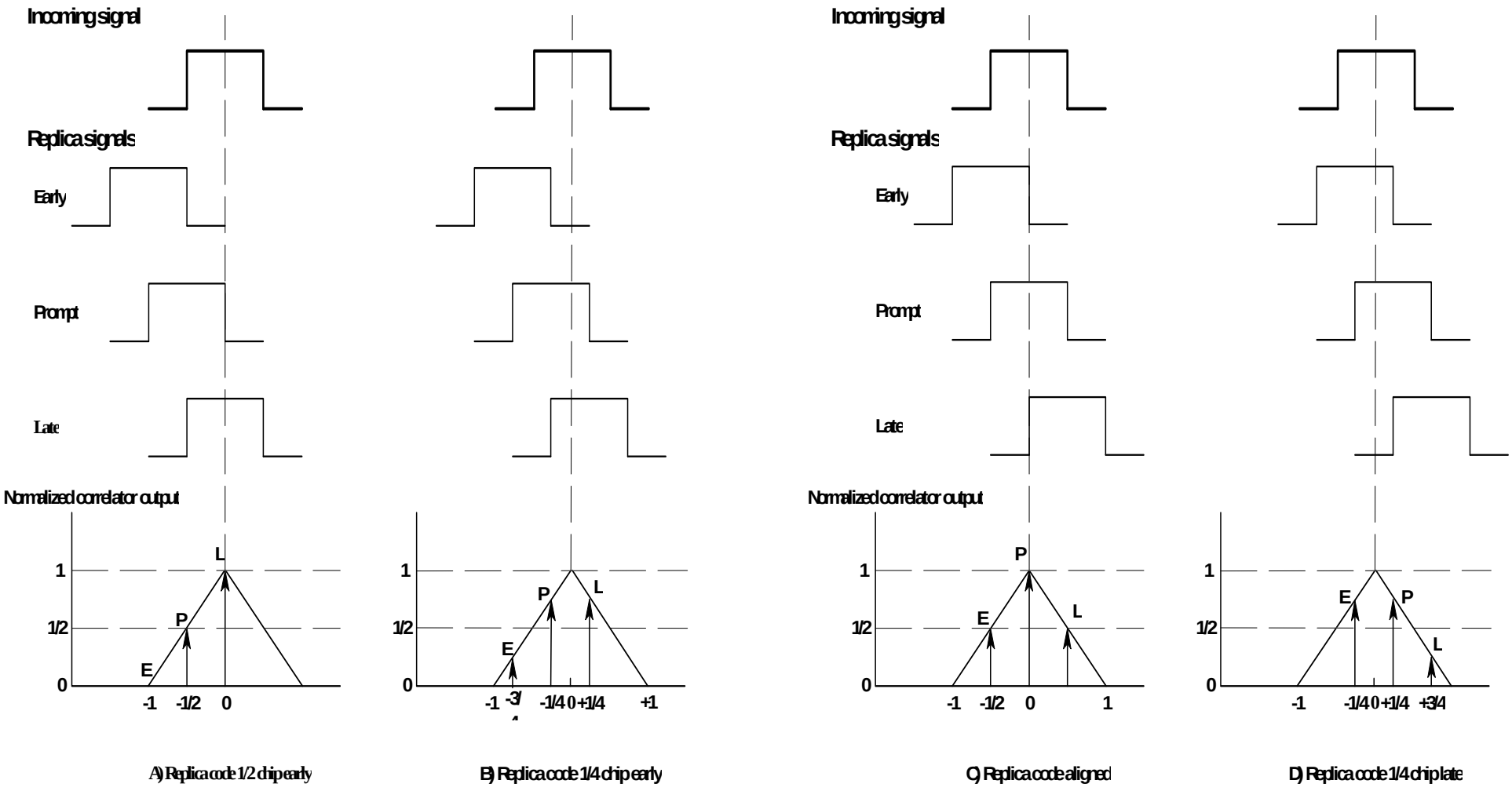


Code correlation process for three different replica code phases

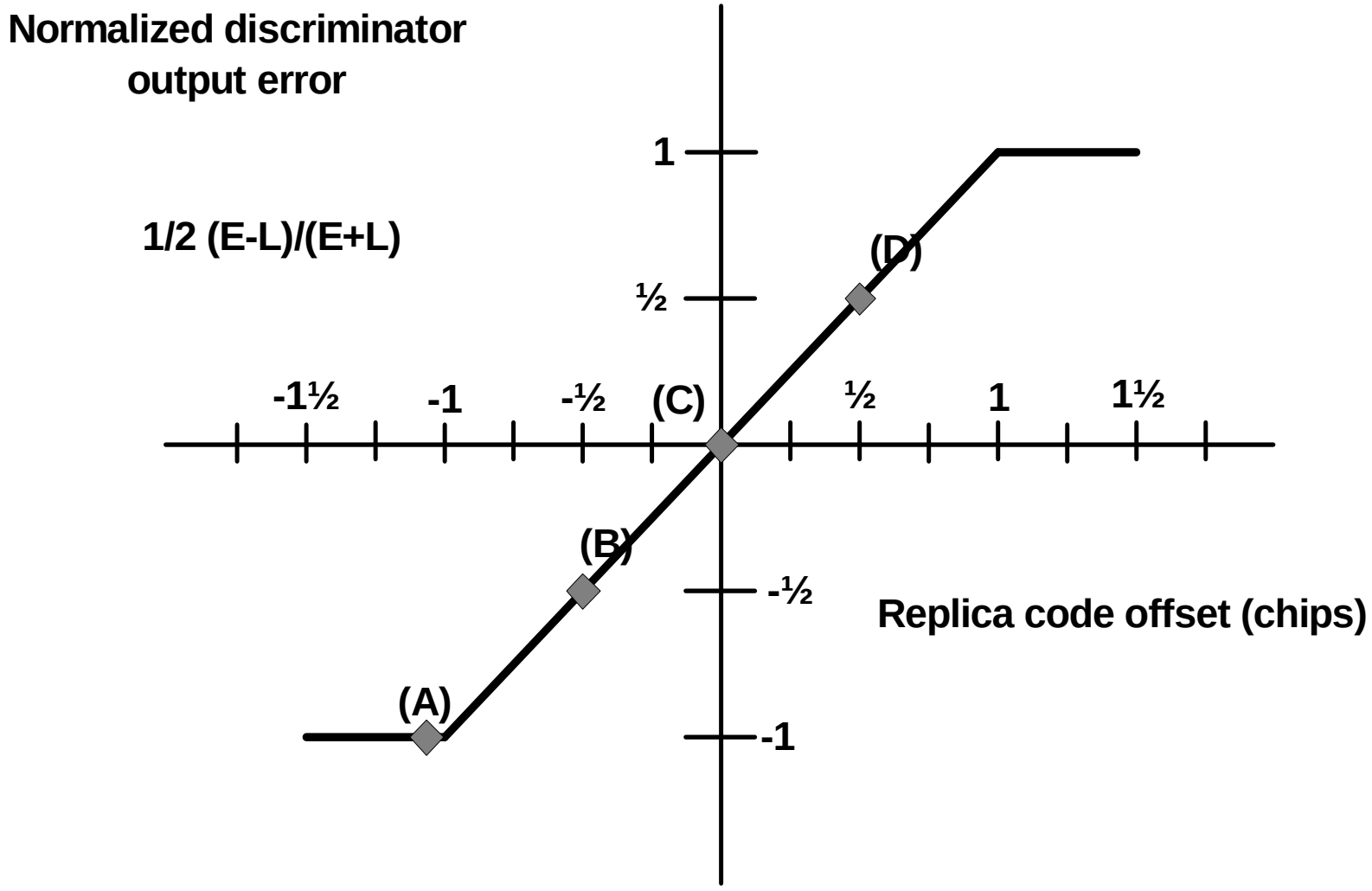




Correlation for replica codes: 1/2 chip early, 1/4 chip early, aligned and 1/4 chip late



Code discriminator output versus replica code offset



Session IV