บทที่ 2

ไทมมิ่งริคัฟเวอรี

ในบทนี้จะอธิบายถึงความสำคัญของไทมมิ่งริคัฟเวอรี (timing recovery) พร้อมทั้งอธิบายหลักการทำงานของไทมมิ่งริคัฟเวอรีแบบที่ใช้กันทั่วไป (conventional timing recovery) ซึ่งอยู่บนพื้นฐานของวงจรเฟสล็อกลูป (PLL: phased-lock loop) นอกจากนี้ยังอธิบายถึง วิธีการออกแบบค่าพารามิเตอร์ของวงจรเฟสล็อกลูป โดยใช้แบบจำลองการทำงานของวงจรเฟสล็อกลูปแบบเชิงเส้น (linearized PLL model) รวมทั้งแสดงผลการทดลองเพื่อให้เห็นถึงความสำคัญของระบบไทมมิ่งริคัฟเวอรี

2.1 บทนำ

ในปัจจุบันนี้ความเจริญก้าวหน้าทางด้านเทคโนโลยีการสื่อสาร (communication) ได้พัฒนาไปอย่างรวดเร็ว โดยเฉพาะอย่างยิ่งการสื่อสารดิจิทัลที่ใช้ในระบบต่างๆ เช่น ระบบโทรศัพท์เคลื่อนที่ และระบบการประมวลผลสัญญาณภายในอุปกรณ์ฮาร์ดดิสก์ไดรฟ์ เป็นต้น เพราะฉะนั้น ความต้องการที่จะเพิ่มประสิทธิภาพโดยรวมของระบบจึงเป็นสิ่งที่จำเป็นมาก เพราะว่าจะได้สามารถรับและส่งข้อมูลได้อย่างน่าเชื่อถือมากยิ่งขึ้น

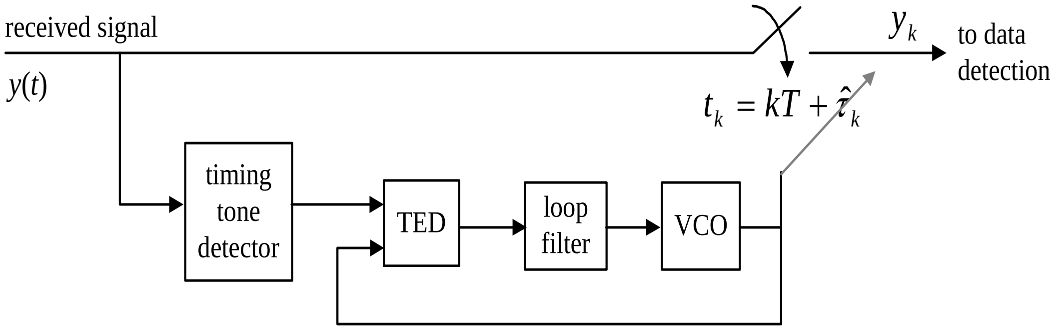
ในระบบสื่อสารดิจิทัล (digital communication system) การส่งสัญญาณจากต้นทางไปยังปลายทาง อุปกรณ์ต้นทางจะทำหน้าที่เปลี่ยนสัญญาณดิจิทัล (digital) ให้เป็นสัญญาณแอนะล็อก (analog) ก่อนที่จะถูกส่งออกไปยังปลายทาง เมื่อสัญญาณแอนะล็อกมาถึงที่อุปกรณ์ปลายทาง ก็จะถูกส่ง

ไปยังวงจรชักตัวอย่าง (sampler) เพื่อทำการแปลงสัญญาณแอนะล็อกให้กลับไปเป็นสัญญาณดิจิทัลในรูปของ "ข้อมูลวิยุต (discrete data)" หรือที่เรียกกันว่า "แซมเปิล (sample)" การชักตัวอย่าง (sampling) สัญญาณแอนะล็อกที่ผิดจังหวะจะทำให้เกิดผลเสียหายอย่างมากกับประสิทธิภาพโดยรวมของระบบ ไทมมิ่งริคัฟเวอรีจะทำหน้าที่ในการเข้าจังหวะ (synchronize) วงจรชักตัวอย่างกับสัญญาณแอนะล็อกที่ได้รับ เพื่อให้ได้ข้อมูลแซมเปิลที่ดีที่สุดออกมา ก่อนที่จะนำข้อมูลแซมเปิลเหล่านั้นไปทำการประมวลผลขั้นต่อไป เช่น ส่งต่อไปยังอีควอไลเซอร์ (equalizer) และวงจรถอดรหัส (decoder) เป็นต้น ดังนั้นไทมมิ่งริคัฟเวอรีจึงนับได้ว่าเป็นองค์ประกอบที่ความสำคัญมากอย่างหนึ่งในระบบสื่อสารดิจิทัล เพราะว่า ถ้าวงจรชักตัวอย่างทำงานไม่ดี ก็จะส่งผลทำให้ข้อมูลแซมเปิลที่ได้ไม่มีคุณภาพหรือมีข้อผิดพลาดมาก และเมื่อส่งข้อมูลแซมเปิลเหล่านี้ไปยังอีควอไลเซอร์และวงจรถอดรหัส ก็จะทำให้ผลลัพธ์ที่ได้มีข้อผิดพลาดมาก นั่นคือ อัตราข้อผิดพลาดบิต (BER: bit-error rate) ของระบบจะมีค่าสูง ซึ่งเป็นสิ่งที่ควรจะหลีกเลี่ยง

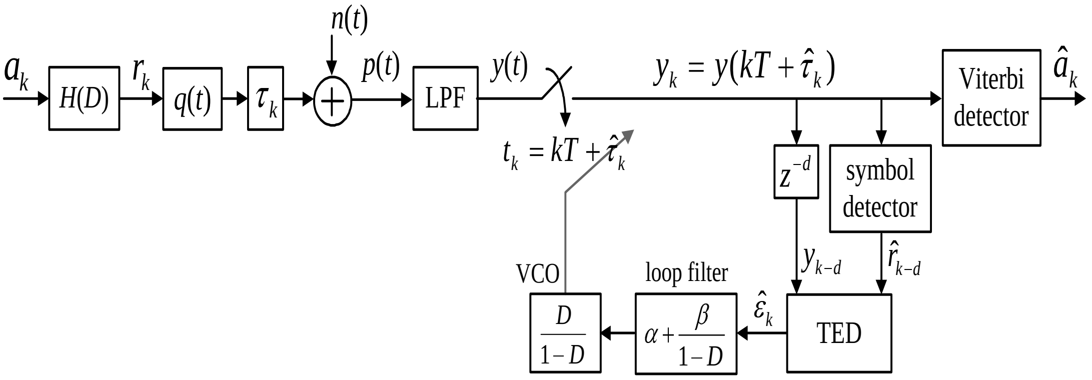
โดยทั่วไปไทมมิ่งริคัฟเวอรีจะทำงานอยู่บนพื้นฐานของวงจรเฟสล็อกลูป (PLL) [4] ซึ่งประกอบไปด้วย วงจรตรวจหาข้อผิดพลาดทางเวลา (TED: timing error detector), วงจรกรองลูป (loop filter), และวงจร VCO (voltage-controlled oscillator) ในทางปฏิบัติ โครงสร้างของไทมมิ่งริคัฟเวอรีจะมีอยู่ 2 รูปแบบ คือ แบบนิรนัย (deductive) และแบบอุปนัย (inductive) โดยจะขึ้นอยู่กับว่า ข้อมูลทางเวลาถูกดึงออกมาใช้ ณ ตำแหน่งก่อนหรือหลังวงจรชักตัวอย่าง ไทมมิ่งริคัฟเวอรีแบบนิรนัย (deductive timing recovery) จะดึงข้อมูลทางเวลา ที่เรียกกันว่า "timing tone" [16] จากสัญญาณแอนะล็อกที่ด้านขาเข้าของวงจรชักตัวอย่าง ตามรูปที่ 2.1 โดยที่ วงจร PLL ถูกนำใช้เพื่อลดผลกระทบของไทมมิ่งจิตเตอร์[1] (timing jitter) ที่แฝงอยู่ในสัญญาณแอนะล็อก ในขณะที่ ไทมมิ่งริคัฟเวอรีแบบอุปนัย (inductive timing recovery) จะใช้วงจร PLL แบบป้อนกลับ (feedback) เพื่อทำหน้าที่ในการดึงข้อมูลทางเวลาจากสัญญาณแอนะล็อกที่ด้านขาออกของวงจรชักตัวอย่าง ดังที่แสดงในรูปที่ 2.2 ข้อดีของไทมมิ่งริคัฟเวอรีแบบอุปนัย คือ ทุกส่วนประกอบของไทมมิ่งริคัฟเวอรีแบบนี้สามารถที่จะถูกสร้างให้เป็นแบบดิจิทัลได้ เนื่องจาก ไทมมิ่งริคัฟเวอรีแบบอุปนัยเป็นที่นิยมใช้งานกันมาก ดังนั้น ในหนังสือเล่มนี้จะเรียกไทมมิ่งริคัฟเวอรีแบบนี้ว่า "ไทมมิ่งริคัฟเวอรีแบบที่ใช้กันทั่วไป (conventional timing

---

[1]ไทมมิ่งจิตเตอร์ คือ สัญญาณรบกวนภายในวงจรชักตัวอย่าง ซึ่งจะทำให้จังหวะของเวลาที่จะทำการชักตัวอย่างแต่ละครั้งเกิดความคลาดเคลื่อนไปจากตำแหน่งที่ต้องการ



รูปที่ 2.1: ไทมมิ่งริคัฟเวอรีแบบนิรนัย



รูปที่ 2.2: แบบจำลองช่องสัญญาณอุดมคติ พร้อมกับไทมมิ่งริคัฟเวอรีแบบอุปนัย

recovery)" สำหรับรายละเอียดเกี่ยวกับไทมมิ่งริคัฟเวอรีแบบนิรนัยสามารถศึกษาเพิ่มเติมได้จาก [20]

2.2 ไทมมิ่งริคัฟเวอรีแบบที่ใช้กันทั่วไป

พิจารณาแบบจำลองช่องสัญญาณอุดมคติ (ideal channel model) ในรูปที่ 2.2 ลำดับข้อมูลอินพุต $a_k \in \{\pm 1\}$ ซึ่งมีคาบเวลาของบิต T ถูกส่งผ่านไปยังช่องสัญญาณ $H(D) = \sum_{k=0}^{\nu} h_k D^k$ โดยที่ h_k คือ ค่าสัมประสิทธิ์ตัวที่ k ของช่องสัญญาณ, D คือ ตัวดำเนินการหน่วงเวลา (delay operator), และ

ν คือ หน่วยความจำของช่องสัญญาณ ดังนั้น สัญญาณ read-back สามารถเขียนเป็นสมการได้ คือ

$$p(t) = \sum_k r_k q(t - kT - \tau_k) + n(t) \tag{2.1}$$

โดยที่ $r_k = \sum_i a_{k-i} h_i$ คือ ข้อมูลเอาต์พุตของช่องสัญญาณที่ปราศจากสัญญาณรบกวน, $q(t) = \sin(\pi t/T)/(\pi t/T)$ คือ ฟังก์ชันซิงก์ (sinc function) หรือสัญญาณพัลส์ไนควิตส์ (Nyquist pulse) ที่มีแบนด์วิดท์เกินเป็นศูนย์ (zero-excess-bandwidth) [16], τ_k คือ ออฟเซตทางเวลาที่ไม่ทราบค่า (unknown timing offset) ตัวที่ k, และ $n(t)$ คือ สัญญาณรบกวนเกาส์สีขาวแบบบวก (AWGN: additive white Gaussian noise) ที่มีความหนาแน่นสเปกตรัมกำลัง (power spectrum density) แบบสองด้านเท่ากับ $N_0/2$ ในหนังสือนี้ ออฟเซตทางเวลา τ_k จะถูกจำลองให้มีลักษณะเป็น "การเดินแบบสุ่ม[2] (random walk)" ซึ่งนิยามโดย [21]

$$\tau_{k+1} = \tau_k + w_k \tag{2.2}$$

เมื่อ w_k คือ ตัวแปรสุ่มเกาส์เซียนแบบ $i.i.d.$ (independent and identically distributed) ที่มีค่าเฉลี่ย (mean) เท่ากับค่าศูนย์ และมีค่าความแปรปรวน (variance) เท่ากับ σ_w^2 หรือเขียนแทนได้ด้วย $w_k \sim \mathcal{N}(0, \sigma_w^2)$ โดยค่า σ_w จะเป็นตัวกำหนดระดับความรุนแรงของไทมมิ่งจิตเตอร์

ที่วงจรภาครับ สัญญาณ read-back จะถูกส่งผ่านไปยังวงจรกรองผ่านต่ำ[3] (LPF) ที่มีผลตอบสนองอิมพัลส์เท่ากับ $q(t)/T$ นั่นคือ มีความถี่ตัด (cut-off frequency) เท่ากับ $1/(2T)$ เพื่อทำหน้าที่กำจัดสัญญาณรบกวนที่อยู่นอกแถบความถี่ (out-of-band noise) จากนั้น ก็จะทำการชักตัวอย่างสัญญาณ read-back ที่เวลา $kT + \hat{\tau}_k$ เพื่อทำให้ได้เป็นข้อมูลแซมเปิล

$$y_k = y(kT + \hat{\tau}_k) = \sum_i r_i q(kT + \hat{\tau}_k - iT - \tau_i) + n_k \tag{2.3}$$

---

[2]แบบจำลองการเดินแบบสุ่มนี้ถูกนำใช้ เพราะว่า เป็นแบบจำลองที่ง่าย และสามารถใช้แทนลักษณะของช่องสัญญาณต่างๆ ได้ง่าย โดยการเปลี่ยนแปลงค่าพารามิเตอร์ σ_w^2 เพียงตัวเดียวเท่านั้น

[3]สำหรับระบบที่มีแถบความถี่จำกัด (band-limited system) นั่นคือ พลังงานของสัญญาณจะถูกจำกัดให้อยู่ในช่วงแถบความถี่ $|f| \leq 1/(2T)$ วงจรกรองผ่านต่ำจะทำให้ข้อมูลเอาต์พุตที่ได้มีค่าทางสถิติที่พอเพียง (sufficient statistic) [23] เหมือนกับการใช้วงจรกรองเหมาะสุด (matched filter) [16]

เมื่อ $\hat{\tau}_k$ คือ ค่าประมาณของ τ_k หรือที่เรียกว่า ออฟเซตทางเฟส (phase offset) ตัวที่ k ของการชักตัวอย่าง, และ n_k คือ AWGN ที่มีค่าเฉลี่ยเท่ากับค่าศูนย์ และค่าความแปรปรวนเท่ากับ $\sigma_n^2 = N_0/(2T)$ นั่นคือ $n_k \sim \mathcal{N}(0, \sigma_n^2)$

วงจร TED จะทำหน้าที่ในการคำนวณหาค่าประมาณของข้อผิดพลาดทางเวลา (timing error) $\epsilon_k = \tau_k - \hat{\tau}_k$ ซึ่งก็คือ ค่าความไม่ตรงกันระหว่างเฟสของสัญญาณแอนะล็อกที่ได้รับกับเฟสของสัญญาณนาฬิกาของวงจร PLL ในทางปฏิบัติแล้ว วงจร TED มีหลายประเภท [4] โดยจะขึ้นอยู่กับลักษณะการนำข้อมูลที่ด้านขาเข้าของวงจร TED มาใช้งาน ซึ่งโดยทั่วไปแล้ว ประสิทธิภาพของไทมมิ่งริคัฟเวอรีจะขึ้นอยู่กับคุณภาพของวงจร TED ในหนังสือเล่มนี้จะพิจารณาเฉพาะวงจร TED ที่นิยมใช้งานกัน นั่นคือ วงจร TED แบบ Mueller and Müller หรือเรียกสั้นๆ ว่า M&M TED [24] ซึ่งจะคำนวณหาค่าประมาณของข้อผิดพลาดทางเวลา ดังนี้

$$\hat{\epsilon}_k = K_T\{y_k\hat{r}_{k-1} - y_{k-1}\hat{r}_k\} \tag{2.4}$$

โดยที่ $\hat{r}_k$ คือ ค่าประมาณของ r_k, K_T คือ ค่าคงตัว (constant) ที่ถูกใช้เพื่อทำให้มั่นใจได้ว่า $E[\hat{\epsilon}_k|\epsilon] = \epsilon$ เมื่อระดับอัตราส่วนค่ากำลังเฉลี่ยของสัญญาณที่ต้องการต่อค่ากำลังเฉลี่ยของสัญญาณรบกวน (SNR: signal-to-noise ratio) มีค่าสูง หรืออาจจะกล่าวได้ว่า ค่า K_T ถูกนำมาใช้เพื่อทำให้ความชันของเส้นโค้งรูปตัวเอส (S-curve) [4] มีค่าเท่ากับค่าหนึ่ง ณ จุดกำเนิด, และ $E[\cdot]$ คือ ตัวดำเนินการค่าคาดหมาย (expectation operator) [10, 25, 26] จากสมการ (2.4) จะเห็นได้ว่า ประสิทธิภาพของวงจร TED จะขึ้นอยู่กับค่าตัดสินใจ (decision) $\{\hat{r}_k\}$ ดังนั้น ประสิทธิภาพของไทมมิ่งริคัฟเวอรีจะเป็นฟังก์ชันของความน่าเชื่อถือของค่าตัดสินใจและ SNR ที่ใช้ จึงเป็นเหตุผลว่า ทำไมวงจรตรวจหาสัญลักษณ์ (symbol detector) ที่ใช้ในไทมมิ่งริคัฟเวอรี คือ วงจรตรวจหาวีเทอร์บิ[4] (Viterbi detector) [15] ที่มีปริมาณการหน่วงเวลาสำหรับการตัดสินใจ (decision delay) เท่ากับ dT หน่วย (เช่น $d = 4$) แทนการใช้งานวงจรตรวจหาขีดเริ่มเปลี่ยนแบบหลายระดับ (multi-level threshold detector) [27]

หลังจากนั้น ค่าประมาณของข้อผิดพลาดทางเวลา $\hat{\epsilon}_k$ จะถูกส่งผ่านไปยังวงจรกรองลูป เพื่อกำจัดสัญญาณรบกวนที่แฝงอยู่ในสัญญาณข้อผิดพลาดทางเวลา และออฟเซตทางเฟสของการชักตัวอย่าง (sampling phase offset) ตัวถัดไป $\hat{\tau}_{k+1}$ ก็จะถูกปรับค่า (update) โดยวงจร PLL อันดับที่สอง

---

[4]ศึกษารายละเอียดหลักการทำงานของวงจรตรวจหาวีเทอร์บิ ได้ในบทที่ 4

(second-order PLL) ตามความสัมพันธ์ดังนี้ [4]

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \beta\hat{\epsilon}_k, \quad (2.5)$$

$$\hat{\tau}_{k+1} = \hat{\tau}_k + \alpha\hat{\epsilon}_k + \hat{\theta}_{k+1} \quad (2.6)$$

เมื่อ $\hat{\theta}_k$ คือ ค่าประมาณของข้อผิดพลาดทางความถี่ (frequency error) [27], และ α และ β คือ พารามิเตอร์ของวงจร PLL [4] ซึ่งจะเป็นตัวกำหนด แบนด์วิดท์ของลูป (loop bandwidth) และอัตรา การลู่เข้า (convergence rate) กล่าวคือ ถ้าค่าพารามิเตอร์ของวงจร PLL ยิ่งมาก แบนด์วิดท์ของลูปก็จะกว้าง ซึ่งจะทำให้อัตราการลู่เข้าก็จะเร็ว แต่สัญญาณรบกวนที่เข้ามาในวงจร PLL ก็จะมีมาก สำหรับในกรณีที่ ระบบมีเฉพาะข้อผิดพลาดทางเฟส (phase error) เท่านั้น วงจรภาครับอาจจะนำวงจร PLL อันดับที่หนึ่ง (first-order PLL) มาใช้งานแทนวงจร PLL อันดับที่สองก็ได้ โดยที่ ออฟเซตทางเฟสของการชักตัวอย่างตัวถัดไป $\hat{\tau}_{k+1}$ จะถูกปรับค่าตามความสัมพันธ์ดังนี้ [4]

$$\hat{\tau}_{k+1} = \hat{\tau}_k + \alpha\hat{\epsilon}_k \quad (2.7)$$

โดยทั่วไป ไทมมิ่งริคัฟเวอรีจะทำงานเป็น 2 ภาวะ (mode) คือ

1) ภาวะการได้มา (acquisition mode) จะทำงานในตอนเริ่มต้นของกระบวนการเข้าจังหวะด้วยความช่วยเหลือของแบบข้อมูล (data pattern) ที่เรียกว่า “preamble” [27] เนื่องจาก ไทมมิ่งริคัฟเวอรีรู้แน่นอนว่า preamble มีลักษณะเป็นอย่างไร จึงทำให้สามารถทราบได้ว่า ค่า $\hat{r}_k$ ที่ถูกต้องคือค่าอะไร ดังนั้นในช่วงภาวะการได้มานี้ วงจร PLL จะใช้ค่า $\hat{r}_k = r_k$ ในการคำนวณหาค่า $\hat{\epsilon}_k$ ตามสมการ (2.4) (วงจรตรวจหาสัญลักษณ์ที่ใช้ในไทมมิ่งริคัฟเวอรีจะยังไม่ถูกใช้งานในช่วงนี้) ซึ่งจะทำให้ได้ค่าที่ถูกต้อง เพราะฉะนั้น กระบวนการไทมมิ่งริคัฟเวอรีในช่วงนี้จึงมีความน่าเชื่อถือมาก โดยจุดประสงค์หลักของภาวะการได้มา ก็คือ การหาค่าประมาณเริ่มต้นของออฟเซตทางเฟสและออฟเซตทางความถี่ (frequency offset) ที่แฝงอยู่ในสัญญาณแอนะล็อกที่จะทำการชักตัวอย่าง

2) ภาวะการติดตาม (tracking mode) จะทำงานต่อจากภาวะการได้มา โดยในขั้นตอนนี้ค่า $\hat{r}_k$ ที่ใช้ในการคำนวณหาค่า $\hat{\epsilon}_k$ ตามสมการ (2.4) จะได้มาจากวงจรตรวจหาสัญลักษณ์ที่ใช้ในไทมมิ่งริคัฟเวอรี (ซึ่งอาจจะมีคุณภาพไม่ดี เมื่อเทียบกับการใช้ r_k จริงๆ) ดังนั้น จุดประสงค์หลักของ