

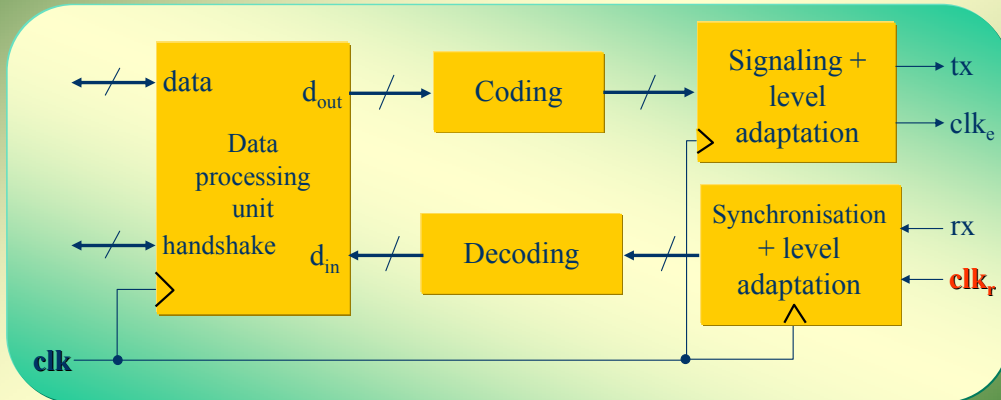
Chapter 2. Timing and synchronisation

2.1. Tracking and recovering clock signals

2.2. Distribution of clock signals

2.1. Tracking and recovering of clock signals

➤ Organisation of a communications system:



2.1. Tracking and recovering clock signals

La imatge mostra l'organització general d'un sistema de comunicacions. El primer lloc la unitat de tractament de dades estarà composta per tots els subsistemes combinacionals (datapath) i seqüencials (unitat de control) necessaris per dur a terme una aplicació específica. Aquesta unitat de dades s'haurà de comunicar habitualment amb altres dispositius, i per tant serà necessari que el sistema disposi de components que permetin establir qualsevol tipus de comunicació (emissió i recepció) amb un sistema remot.

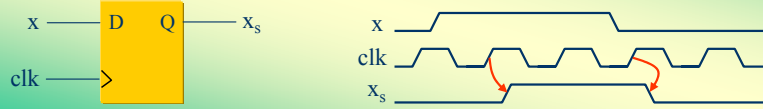
Pel que fa al procés d'emissió, les dades produïdes per la unitat de tractament habitualment s'hauran de codificar d'alguna manera per tal d'adaptar-se el protocol de comunicació establert. Aquest procés de codificació s'implementa habitualment de forma combinacional. Una vegada les dades estan codificades s'hauran d'ordenar temporalment, i a continuació s'hauran d'ajustar a un determinat protocol de senyalització. Finalment serà necessari adaptar els nivells físics de sortida, d'acord amb les especificacions del protocol utilitzat. Per tan, com a resultat es disposarà de un senyal de dades de sortida i, de forma opcional, d'un senyal de rellotge.

Respecte al procés de recepció, el nostre sistema disposarà com a entrada d'un senyal de dades generat per un dispositiu extern, i de forma opcional d'un senyal de rellotge generat també per aquest dispositiu. Donat que aquests senyal s'han generat externament, són asíncrons pel nostre sistema, de manera que el primer que caldrà fer, una vegada recuperats els nivells de tensió adients, serà sincronitzar-los. Una vegada s'hagin obtingut les dades sincronitzades, s'hauran de descodificar (procés invers a la codificació) abans que aquestes puguin ser utilitzades per la unitat de tractament.

Com es pot apreciar a la imatge, és important tenir present que tots els elements seqüencials del nostre sistema treballaran amb un únic senyal de rellotge, que serà sempre diferent al possible rellotge que acompanyi les dades en mode recepció.

• Synchronisation:

- Establishment or recognition of a specific temporal ordering in the changes of a signal



➤ Utility:

- Arbitration of asynchronous requests (bus protocols)
- Sampling of asynchronous signals with a clock signal
- Transmission of asynchronous signals between systems controlled by independent clock signals

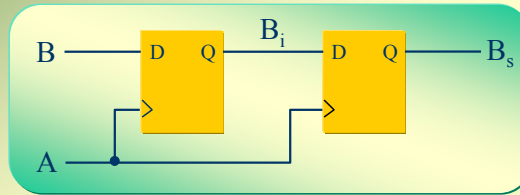
2.1. Tracking and recovering clock signals

Com s'ha comentat a la transparència anterior, la sincronització és un procés fonamental a qualsevol sistema de comunicació.

En mode emissió, la sincronització implica imposar una determinada ordenació temporal a les dades que s'estan enviant. Aquesta ordenació es realitzarà a partir del senyal de rellotge del sistema.

Pel que fa a la recepció, la sincronització implica reconèixer la ordenació temporal de les dades que es va imposar al l'element emissor. Aquest reconeixement implica, tal i com es mostra a la figura de la part superior dreta de la imatge, convertir els canvis del senyal que s'està rebent en canvis interns que estiguin governats pel senyal de rellotge del sistema.

• Example:



- $\tau_s = 200$ ps, $t_a = 200$ ps, $f_B = 1$ MHz
- $f_A = 100$ MHz $\Rightarrow t_w = 10$ ns

- $P_e = 3.86 \cdot 10^{-24}$
- $f_e = 3.86 \cdot 10^{-18}$
- $MTBF = 2.59 \cdot 10^{17}$ s

Robust synchronisation with one clock latency (T_A)

2.1. Tracking and recovering clock signals

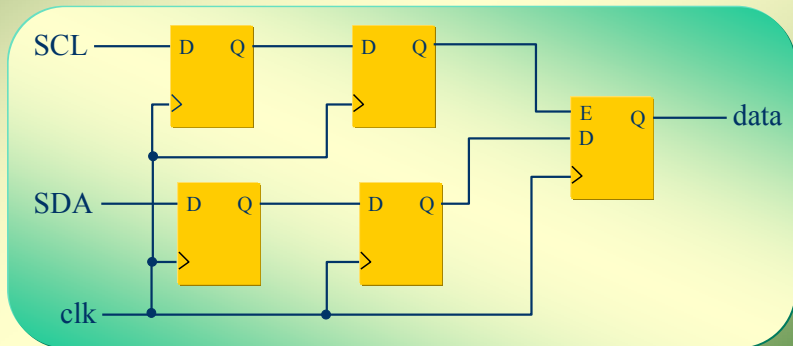
A la part superior de la imatge es mostra el exemple corresponent a una possible solució per evitar errors de sincronització. En aquest exemple s'utilitza un segon registre per esperar a que el sistema hagi arribat a un estat estable. És a dir, amb aquest segon registre el temps d'espera a que el sincronitzador arribi a un estat estable és el període de rellotge del sistema.

A la part esquerra de la imatge es mostren valors habituals pels paràmetres del sistema. En funció d'aquests valors es calculen, a la part dreta de la imatge, els valors corresponents a la probabilitat d'error, a la freqüència d'errors de sincronització i al temps promig entre errors. Cal tenir present que aquest temps promig entre errors correspon a més de 81 bil·lions d'anys. És a dir, el fet d'afegir un registre addicional al sistema encarregat de sincronitzar el senyal de dades permet construir un sincronitzador molt robust.

- **Application example:**

- **Synchronisation of the data line in the I²C protocol:**

- Clock signal (SCL) generated by the *master* device
- Data line (SDA) valid when SCL=1

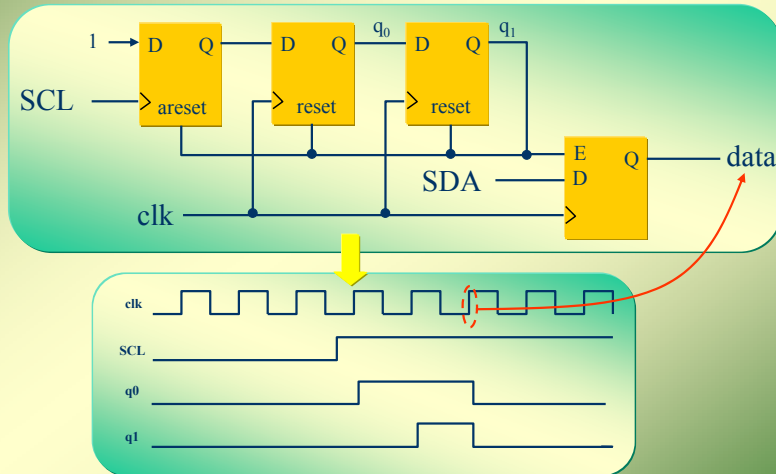


2.1. Tracking and recovering clock signals

A partir dels resultats anteriors, la figura mostra un exemple de com es pot realitzar la sincronització dels senyal de rellotge, SCL, i de dades, SDA, corresponents al protocol de comunicació sèrie I2C.

• Optimisation (I²C):

- Using the SCL line as a clock signal:



2.1. Tracking and recovering clock signals

Aquesta imatge mostra una possible optimització del sistema sincronitzador mostrat a la transparència anterior. Cal tenir present que aquesta optimització només és vàlida pel protocol I²C, doncs en aquest protocol la línia de dades només és vàlida quan el senyal de rellotge, SCL, es troba a nivell '0'.

• Strategies for tracking and recovering clock signals:

- **Goal:** Re-generate (in frequency or phase) a given clock signal

- **Utility:**

- Synchronisation of systems
- Distribution of clock signals within a system

- **Strategies:**

- Clock tracking (PLLs, DLLs)
- Oversampling

2.1. Tracking and recovering clock signals

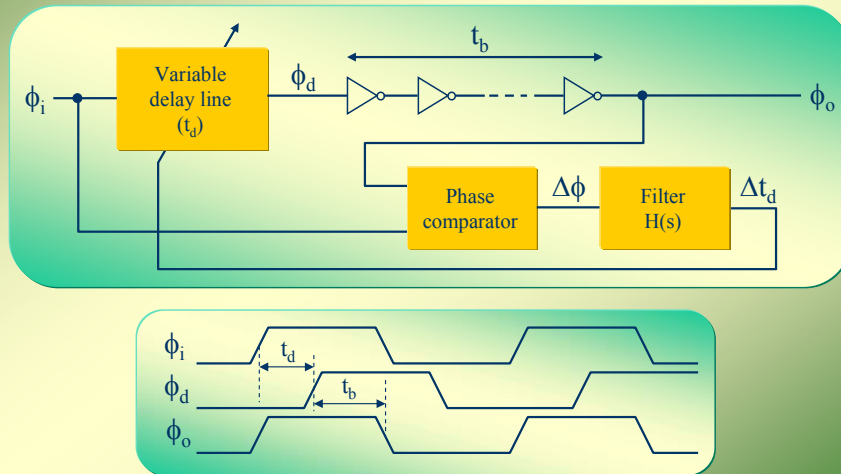
Una vegada considerats tots els aspectes relacionats amb la sincronització del senyal de dades d'un sistema es consideraran a continuació les estratègies disponibles per tal de dur a terme el seguiment o recuperació d'un senyal de rellotge extern.

L'objectiu final d'aquestes estratègies consistirà en regenerar el senyal de rellotge que ha produït l'ordenació temporal al senyal de dades rebut.

Es poden considerar bàsicament dos tipus d'estratègies: Les basades en regenerar de forma explícita un senyal de rellotge i les basades en deduir el senyal de rellotge a partir de les dades rebudes.

El primer grup d'estratègies donarà lloc als components que normalment es coneixen com a PLLs (Phase Locked Loops) i DLLs (Delay Locked Loops), mentre que el segon grup d'estratègies es basaran en dur a terme un sobremostreig del senyal de dades rebut.

• Delay Locked Loop (DLL):



2.1. Tracking and recovering clock signals

La part superior de la imatge mostra la organització general del que s'anomena bucle enganxat en fase o DLL (Delay Locked Loop). L'objectiu d'aquest sistema és compensar el retard entre un senyal de rellotge d'entrada i un senyal de rellotge de sortida. Aquesta compensació, com es mostra a la part inferior de la imatge, es pot aconseguir introduint un retard variable, t_d , que sumat al retard existent entre els dos senyals de rellotge, t_b , faci que aquests estiguin finalment en fase.

Els components bàsics d'un bucle enganxat en fase són:

- Línia de retard variable: Introdueix el retard que s'afegirà al retard global de sistema.
- Comparador de fase: Sistema encarregat de determinar la diferència de fase entre els dos senyals de rellotge.
- Filtre passa baixos: Determina el valor de retard a afegir al senyal de entrada en funció de la diferència de fase entre aquest i el senyal de sortida.

• Loop dynamics:

- Delay line and phase comparator respond instantaneously:

$$\frac{\Delta\phi(s)}{e(s)} = \frac{1}{1+H(s)} \quad H(s) = \frac{k \cdot a}{s+a} \quad \frac{\Delta\phi(s)}{e(s)} = \frac{s+a}{s+a(k+1)}$$

$$e(t) = 0 \Rightarrow \Delta\phi(t) = \Delta\phi(0) \cdot e^{-a \cdot (k+1) \cdot t}$$

$$H(0) \gg 1$$

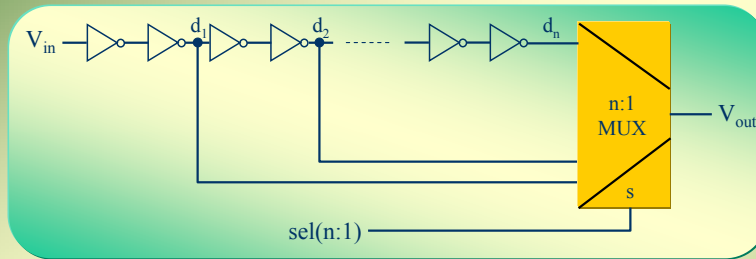
➤ Residual error: $\Delta\phi_r = \frac{2\pi}{T} \cdot \left(\frac{T/2 - t_{d0} - t_b}{1+k_L} \right), k_L = \frac{2\pi}{T} H(0)$

2.1. Tracking and recovering clock signals

Suposant que la línia de retard i el comparador de fase responen de forma pràcticament instantània, a la part superior de la imatge es mostra la funció de transferència del sistema realimentat. Si a aquesta funció de transferència se suposa que el filtre passa baixos té una funció de transferència, $H(s)$, de primer ordre, serà possible arribar a una expressió per a l'error de fase entre els senyals com el mostrat a la part central de la figura.

A partir d'aquesta expressió és possible deduir el valor de l'error residual de fase, que es mostra a la part inferior de la figura. Com es pot deduir fàcilment a partir d'aquesta expressió, per tal que l'error residual de fase tingui el mínim valor possible és convenient que el valor del guany en continua del filtre passa baixos sigui el més gran possible.

• Variable delay line:



• Drawbacks:

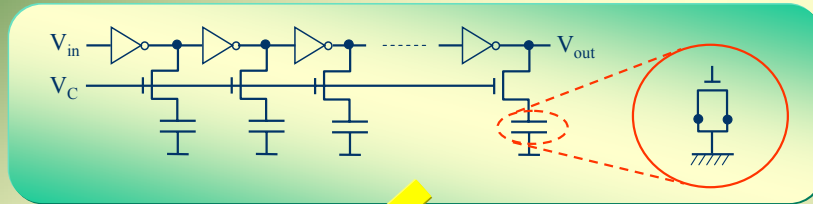
- Large dynamic range and fine tuning imply a large number of stages \Rightarrow Superposition of jitter
- Low accuracy tuning
- Digital control (analog control more compact)

2.1. Tracking and recovering clock signals

La figura mostra una possible estructura per realitzar la línia de retard variable d'un DLL. Com es pot apreciar, es tracta d'un multiplexor que és capaç de produir a la seva sortida versions endarrerides del senyal d'entrada. L'etapa de retard bàsica està constituïda per un parell d'inversors. El principal problema d'aquesta estructura és degut al fet que el retard d'un parell d'inversors és normalment molt petit. Això obliga a que si es desitja compensar un marge d'error de fase molt gran sigui necessari incloure un gran nombre d'etapes, i com a conseqüència es tindrà un nivell de jitter (error de fase) elevat, degut a que serà impossible aconseguir que totes les etapes tinguin el mateix retard.

Per tant, l'ajust d'aquest sistema serà poc precís, i a més la seva implementació tindrà uns requeriments d'àrea elevats, degut a que el control del retard s'efectua de forma digital.

• Delay line controlled by capacitance:



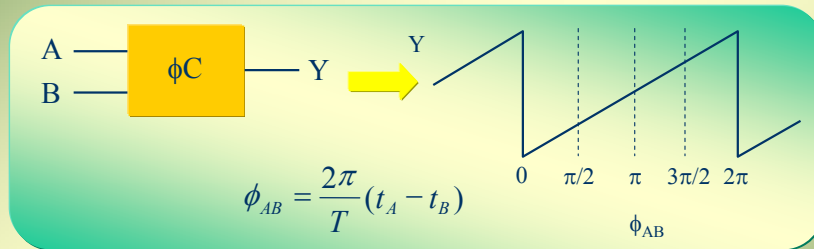
- Transition 0 → 1 in inverter:
 - Capacitor charges to $V_C - V_{TN}$
- Transition 1 → 0 in inverter:
 - Capacitor discharges when the inverter output is below $V_C - V_{TN}$

2.1. Tracking and recovering clock signals

La part superior de la imatge mostra una possible implementació analògica de la línia de retard variable d'un DLL. En aquesta implementació es controla mitjançant una tensió analògica, V_C , el retard de resposta dels inversors que componen la línia de retard. Això es pot aconseguir, tal i com es mostra a la figura, que els inversors triguin més o menys temps a carregar un condensador equivalent situat a la seva sortida.

• Phase comparator:

- **Function:** Measure the relative delay (phase) of two signals

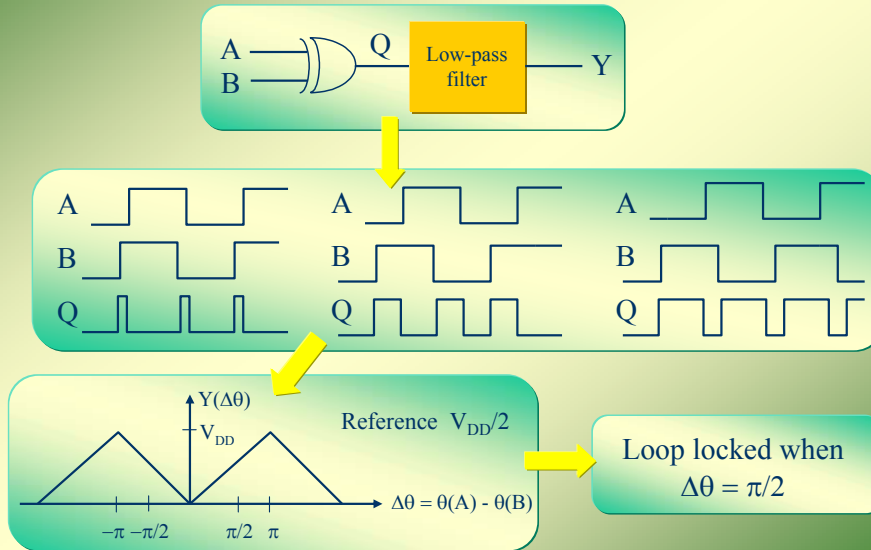


- A and B periodic signals \Rightarrow measure of a relative phase
- A clock signal and B data signal \Rightarrow measure of the relative delay of the transitions of B with respect to the edges of A

2.1. Tracking and recovering clock signals

La funció fonamental del comparador de fase que forma part d'un DLL és determinar la diferència de fase entre dos senyals. A la part superior esquerra de la imatge es mostra el símbol d'un comparador de fase, mentre que a la part superior dreta es representa la seva funció de transferència ideal. Com es pot apreciar, es tracta d'un sistema que ha de proporcionar a la seva sortida un senyal de valor tant més elevat com més elevada sigui la diferència de fase entre els senyals d'entrada.

• Phase comparator based on XOR functions:



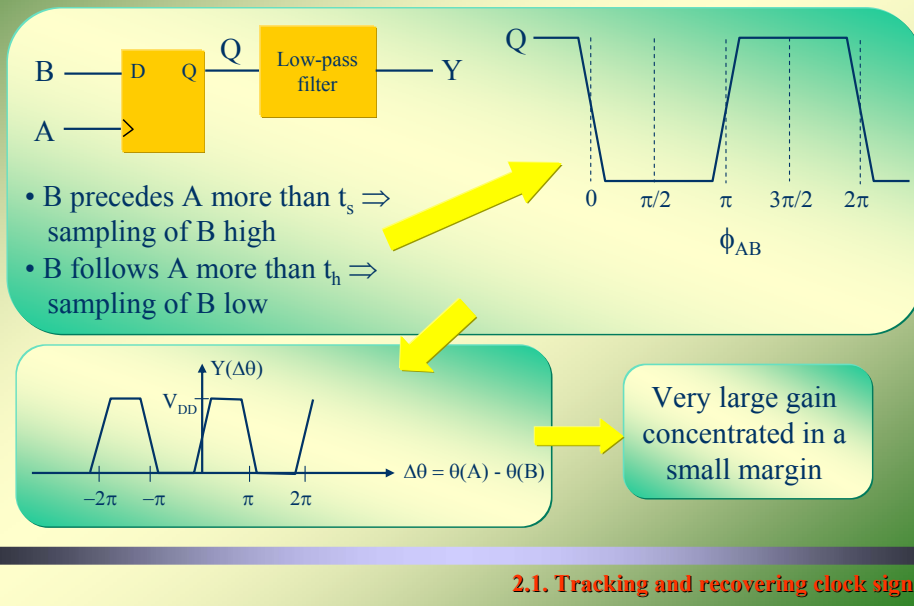
2.1. Tracking and recovering clock signals

Una forma senzilla de realitzar un comparador de fase consisteix a realitzar la funció XOR de dos senyals d'entrada i aplicar a la sortida d'aquesta un filtre passa baixos, tal i com es mostra a la part superior de la imatge.

A la part central de la imatge es mostra la sortida de la funció XOR per diversos valor de diferència de fase entre les seves entrades. A partir d'aquestes gràfiques és fàcil deduir que el filtre passa baixos proporcionarà a la seva sortida un valor més gran quan més gran sigui la diferència de fase entre els senyals d'entrada.

A la part inferior esquerra de la imatge es mostra la funció de transferència d'aquest comparador de fase. Per tal d'aprofitar tot el marge lineal del sistema serà convenient doncs que el bucle estigui enganxat quan la diferència de fase entre els senyals d'entrada sigui d'un quart de període.

• Phase comparator based on registers:

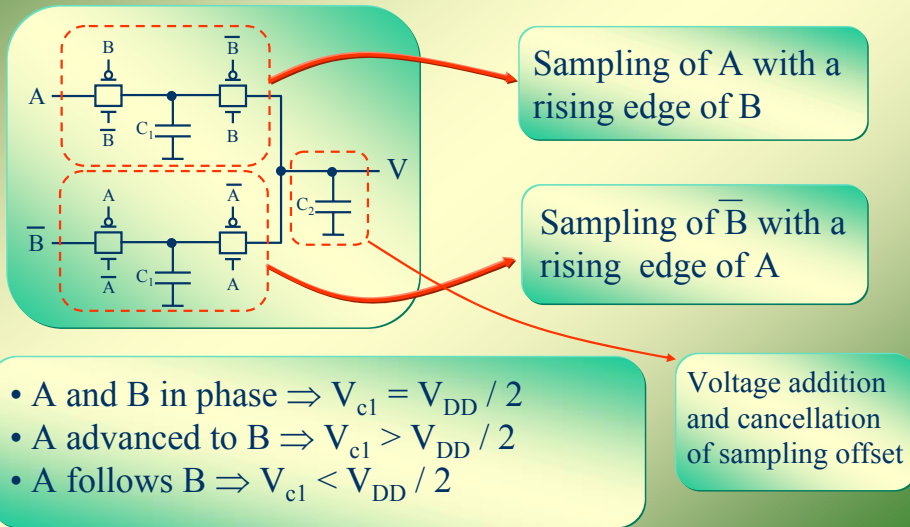


2.1. Tracking and recovering clock signals

Una altra possibilitat per realitzar el comparador de fase d'un DLL consisteix en utilitzar un registre, el qual tindrà connectades les seves entrades de dades i de rellotge als dos senyals la diferència de fase dels quals es tracta de determinar, tal i com es mostra a la part superior esquerra de la imatge.

El principal problema d'un comparador de fase com aquest és degut al fet que la seva zona lineal es troba concentrada, amb un guany molt elevat, a un marge de fase molt estret.

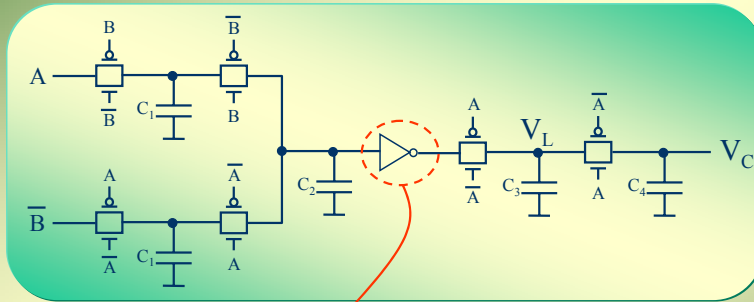
• Analog version of the phase comparator:



2.1. Tracking and recovering clock signals

A la part superior esquerra de la imatge es mostra una alternativa analògica per realitzar un comparador de fase de dos senyals. Com es pot deduir fàcilment, el senyal de sortida, V, d'aquest sistema tindrà un valor més elevat quan més gran sigui la diferència de fase entre els senyals d'entrada.

• Integration of the filter with the phase comparator:



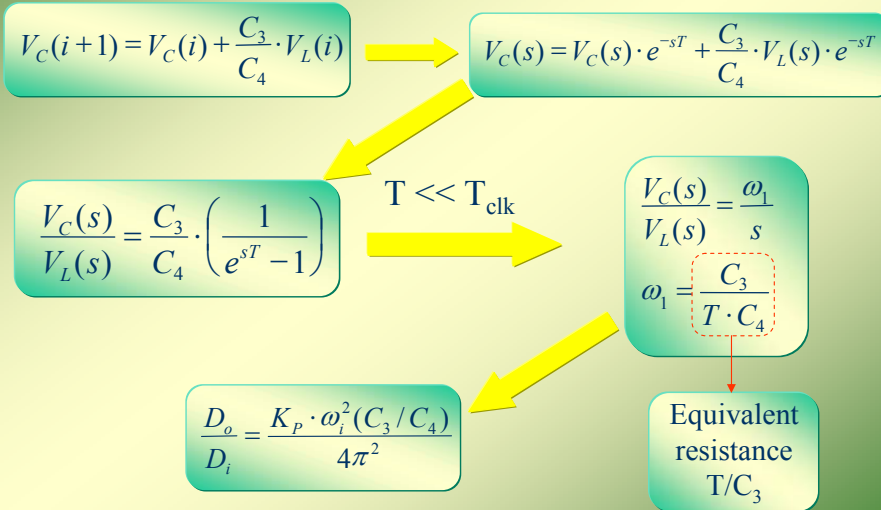
• Gain of the filter:

- A advanced to B \Rightarrow Output high
- B advanced to A \Rightarrow Output low
- Locked loop \Rightarrow Input and output equal $V_{DD}/2$

2.1. Tracking and recovering clock signals

La part superior de la imatge mostra la forma en que es pot construir de forma conjunta un comparador de fase al qual s'integra el filtre passa baixos. La part esquerra de l'esquema és el sistema que s'ha identificat a la transparència anterior com a detector de fase. A continuació se situa un inversor, que serà l'element encarregat de proporcionar un elevat guany en continua (un inversor es pot considerar com un amplificador inversor amb un elevat guany en continua). Finalment, el transistor de pas i les capacitats C_3 i C_4 constitueixen, tal i com es demostrarà a continuació, un filtre passa baixos.

• Analysis:

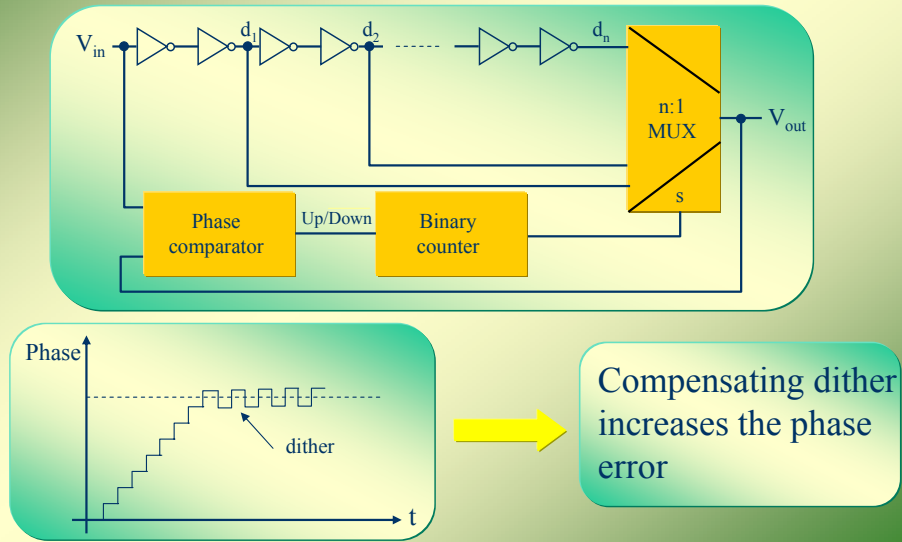


2.1. Tracking and recovering clock signals

Per demostrar que el conjunt transistor de pas més capacitats C_3 i C_4 constitueixen un filtre passa baixos només caldrà plantejar les relacions, en termes de diferències finites, entre les tensions V_C i V_L , tal i com es mostra a la part superior esquerra de la imatge. A partir d'aquesta relació podem determinar la transformada de Laplace, de forma que finalment a la part central de la imatge es pot deduir la funció de transferència del sistema. Suposant que el període del senyals és molt inferior al període de mostreig del sistema, finalment aquesta funció de transferència es pot simplificar de la manera que es mostra a la part central dreta de la imatge.

Aquesta funció de transferència correspon a un filtre passa baixos, i a partir dels seus paràmetres característics es pot deduir que la capacitat C_3 es comporta de forma efectiva com a una resistència equivalent de valor T/C_3 .

• Loop with digital control:



2.1. Tracking and recovering clock signals

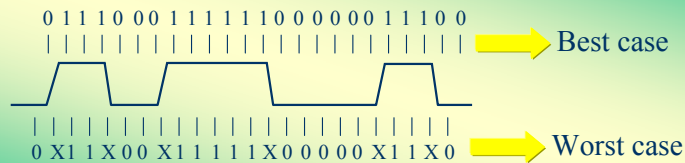
A partir dels sistemes que s'han analitzat amb anterioritat es pot deduir l'estructura d'un bucle amb control digital, tal i com es mostra a la part superior de la imatge. A aquest bucle es fa servir la línia de retard amb control digital que s'ha presentat amb anterioritat. El control del bucle es realitza amb un dels comparadors de fase mostrats anteriorment, la sortida del qual actua com a senyal de control ascendent/descendent d'un comptador binari sincron. La sortida d'aquest comptador actua com a senyal de selecció del multiplexor que tria el valor de retard a afegir al senyal d'entrada.

El problema fonamental d'aquest sistema és degut al fet que en regim permanent hi haurà una fluctuació a la fase del senyal de sortida, tal i com es mostra a la part inferior esquerra de la imatge. Aquesta fluctuació és deguda a la variació al bit menys significatiu del comptador. Si aquesta fluctuació es compensa de forma que el bit menys significatiu del comptador no s'utilitza, llavors l'error de fase en regim permanent serà més elevat.

• Clock recovery by oversampling:

- The clock signal in the receiver has no relation with that of the emitter \Rightarrow A higher clock frequency is used in order to recover the data correctly

Oversampling x3



2.1. Tracking and recovering clock signals

La recuperació de relloige és una tècnica a utilitzar quan el senyal de relloige de l'emissor no estigui disponible de forma explícita. En aquest cas la recuperació del relloige s'haurà de fer a partir del senyal de dades rebut. Donat que el sistema emissor treballa amb un senyal de relloige diferent al del sistema receptor, aquest haurà de mostrejar les dades a una freqüència superior a la corresponent al protocol de comunicació a fi de poder recuperar correctament el senyal de relloige.

La figura de la part inferior de la imatge mostra les dues situacions extremes que es poden donar en un sistema de recuperació per sobremostreig. A la part central de la figura es mostra el senyal de dades que rep el sistema receptor. Si aquest sistema mostreja les dades a una freqüència 3 vegades superior a aquella amb la que aquestes s'han generat, a sobre de la línia de dades es mostra el millor cas d'alineament entre els senyals de relloige a l'emissor i al receptor, de forma que cap de les transicions del senyal de dades està situat dins de l'interval d'obertura de l'element sincronitzador.

A la part inferior del senyal de dades es mostra la situació contrària. En aquest cas tots els canvis al senyal de dades coincidiran dins de l'interval d'obertura de l'element sincronitzador, i per tant donaran lloc a metastabilitat (representada amb un valor X a la figura).

Per tant, tal i com es pot deduir a partir d'aquesta figura, el mecanisme de recuperació del senyal de relloige consistirà en identificar, a partir de les transicions al senyal de dades, la posició dels flancs del senyal de relloige al sistema emissor.

• Considerations on oversampling:

- **Oversampling xN** : A sequence of M 1's (0's) in the emitter is converted into a sequence of length x ,
 $N \cdot M - 1 \leq x \leq N \cdot M + 1$

An **oversampling $> x3$** is able to distinguish sequences of symbols of different length even in the worst case of Temporal delay

Clock recovery by establishing the position of the transitions

Maximum resolution of T_{bit}/N

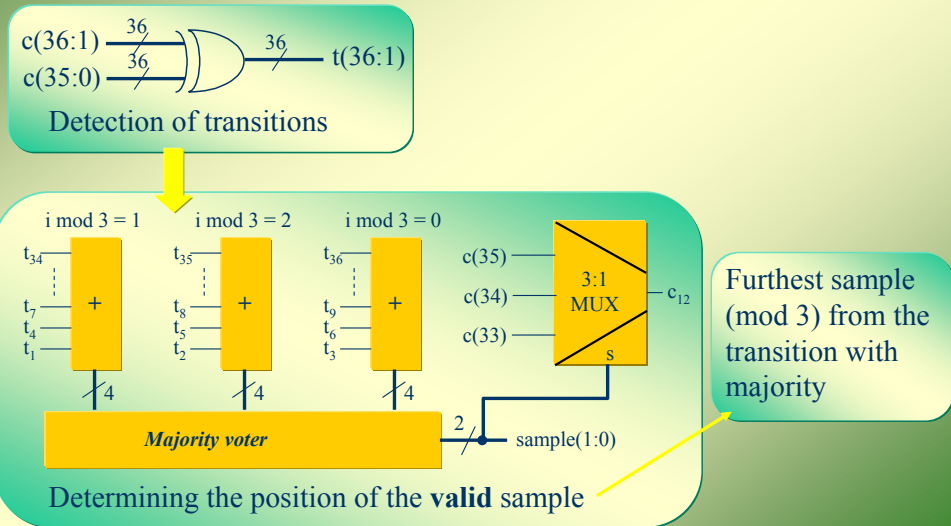
Quantisation jitter

2.1. Tracking and recovering clock signals

A partir de l'exemple que s'ha mostrat a la transparència anterior és fàcil deduir que amb un factor de sobremostreig de com a mínim 3 serà possible recuperar un senyal de rellotge a partir del senyal de dades rebut. Això és degut a que, fins i tot en el pitjor dels casos, es disposarà de dues mostres vàlides per període de bit. Com s'ha comentat anteriorment, la recuperació del rellotge es farà llavors a partir de la detecció de les transicions al senyal de dades, doncs la posició d'aquestes transicions indicarà també la posició dels flancs del senyal de rellotge que les ha generat.

Degut a que la resolució màxima que es pot assolir ve donada pel factor de sobremostreig, hi haurà necessàriament un jitter de quantització superposat al senyal recuperat, que serà més petit quan més gran sigui el factor de sobremostreig.

• Realisation for $N=3$, 12-bit windows:



2.1. Tracking and recovering clock signals

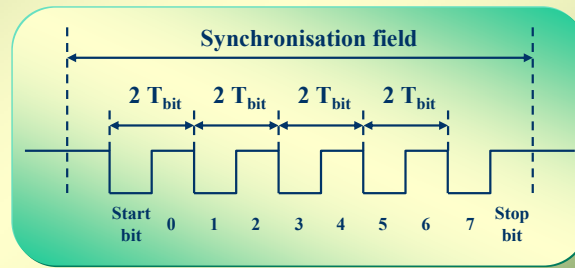
A la imatge es mostra un exemple de la recuperació del senyal de rellotge pel cas que el factor de sobremostreig sigui 3 i la recuperació del senyal es faci a partir de 12 períodes de bit del senyal de dades rebut.

A la part superior de la figura es mostra de forma esquemàtica el subsistema encarregat de detectar les transicions al senyal de dades. Aquest sistema fa la funció XOR de cada mostra del senyal de dades amb l'anterior, de manera que si hi ha una diferència (transició) generarà un valor '1' a la seva sortida. Cal tenir present que a la figura està representat un array de 36 portes XOR, donat que el sistema treballarà amb 36 mostres del senyal de dades, i que les mostres del senyal de dades són els valors $c(0) \dots c(36)$.

A continuació, a la part inferior de la imatge es mostra el sistema utilitzat per detectar quina és la posició de les mostres del senyal de dades on es produeixen més transicions. Donat que hi hauran 3 mostres per període de bit, els valors $t_1, t_4, t_7, \dots, t_{34}$ indicaran si, dins de la finestra utilitzada (12 bits de dades) s'han produït transicions a la primera mostra del senyal de dades. Per tant, fent la suma d'aquests valors s'obté el nombre de transicions que s'han produït a la primera mostra del senyal de dades. El mateix caldrà fer per a la segona i tercera posició de les mostres. A partir d'aquest valor un subsistema votador de majoria s'encarregarà de determinar a quina posició es produeixen més transicions. A partir d'aquesta informació generarà el codi (senyal **mostra(1:0)** a la figura) corresponent a la posició més allunyada d'aquella on estan localitzats majoritàriament els canvis al senyal de dades. D'aquesta manera es podrà garantir el mostreig correcte del senyal de dades.

• Alternative realisation (LIN protocol):

- Use of the synchronisation field with a pattern 0x55 before each data frame
- Use the falling edges of the synchronisation pattern to determine the actual data oversampling rate



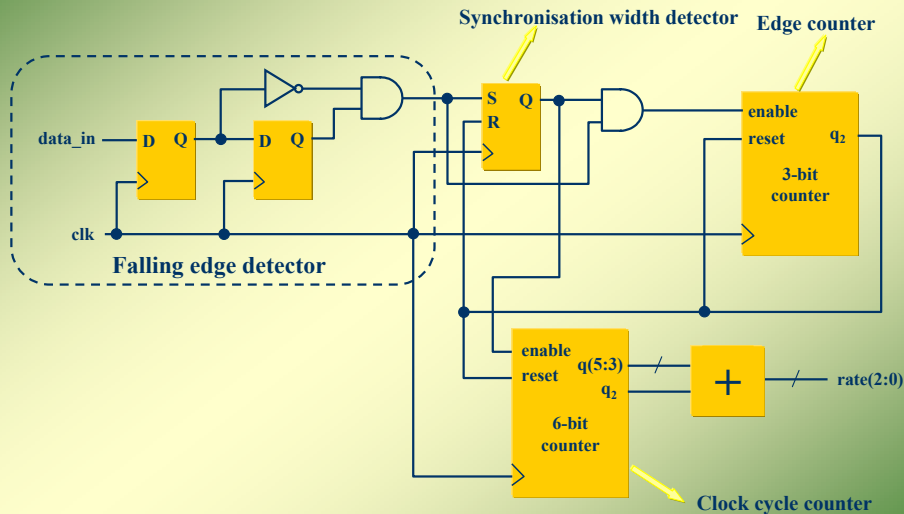
2.1. Tracking and recovering clock signals

Una realització alternativa per recuperar el senyal de rellotge és la que s'utilitza al protocol LIN (Local Interconnect Network - bus sèrie de baixa velocitat utilitzat a la indústria de l'automòbil, tal i com es veurà en detall al mòdul 3 de l'assignatura). Dins d'aquest protocol cada trama de dades porta associat un camp de sincronisme. Dins d'aquest camp de sincronisme, delimitat per un bit de start un bit de stop, s'envia un byte de dades amb valor hexadecimal 0x55 (és a dir, una seqüència alternada de valors lògics '1' i '0'). La figura situada a la part inferior de la imatge mostra la seqüència corresponent a aquest camp de sincronisme. Com es pot deduir fàcilment, exceptuant el bit de start, dins d'aquest camp de sincronisme es produeixen 4 flancs de baixada al senyal de dades.

Per tant, en aquest cas la recuperació del senyal de rellotge (és a dir, la determinació de la posició òptima de mostreig de senyal de dades) consistirà en determinar quants cicles de rellotge del sistema receptor es produeixen dins de l'interval corresponent a aquests 4 flancs de baixada. A partir d'aquest valor es podrà determinar fàcilment el valor del factor de sobremostreig efectiu, que serà directament aquest valor dividit per 4 (és a dir, el valor final del compteig descartant el dos bits menys significatius i afegit - per no tenir problemes d'arrodoniment - el bit menys significatiu descartat).

Una vegada determinat el factor de sobremostreig efectiu es podrà establir quina és la posició òptima per mostrejar el senyal de dades al receptor.

• Realisation (oversampling x4):



2.1. Tracking and recovering clock signals

La imatge mostra el esquema corresponent a un sistema que pot dur a terme la recuperació del senyal de rellotge utilitzant el protocol LIN. El bloc situat a la part superior esquerra de la figura s'encarrega de detectar flancs de baixada al senyal de dades. És a dir, aquest bloc generarà un pols amb una amplada d'un període del senyal de rellotge *clk* cada vegada que es produeixi un flanc de baixada al senyal *data_in*. La sortida d'aquest bloc activarà un registre RS que tindrà com a missió detectar l'interval de temps corresponent als 4 flancs de baixada al senyal de dades. El comptador binari sincron de 3 bits mostrat a la part superior dreta de la imatge s'encarregarà de comptar el nombre de flancs de baixada que s'han produït al senyal de dades, mentre que el comptador binari sincron de 6 bits que es mostra a la part inferior de la imatge determinarà el nombre de cicles de rellotge que s'han produït dins d'aquest interval (es necessitaran 6 bits, doncs per un factor de sobremostreig nominal de 4 el nombre de cicles de rellotge serà 32).

Com es pot apreciar fàcilment, el sumador situat a la part inferior dreta de la figura s'encarrega de sumar el valor de sortida del comptador de 6 bits desplaçat tres posicions (és a dir, dividint per 8) més el bit més significatiu descartat.

Quan el comptador de 3 bits arribi al valor 4 es produirà un reset al detector d'interval de sincronisme i del comptador de 6 bits. En aquest moment la sortida del sistema, **factor(2:0)**, contindrà el valor corresponent al factor de sobremostrig efectiu al sistema receptor.

2.2. Distribution of clock signals

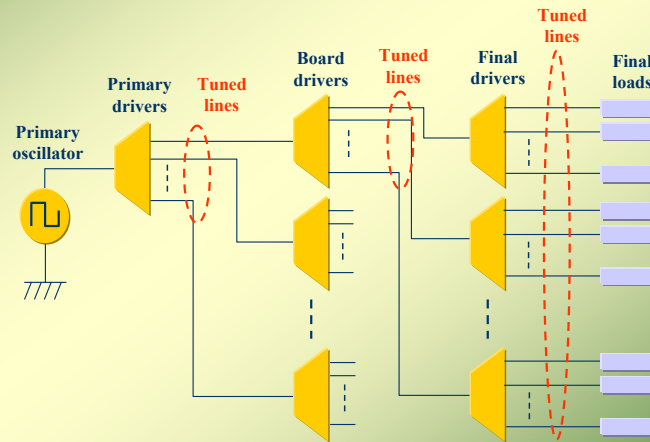
- **Goal:** Distribute a clock signal to all the memory elements in a clock domain with minimum skew and jitter values
- **Skew:** DC component of the phase noise, due to lines of different length or to parameter variations in the components of the system
- **Jitter:** AC component of the phase noise, due to noise in the power supply or to additive noise (crosstalk or intersymbol interference)

2.2. Distribution of clock signals

Finalment, per acabar aquest mòdul, es consideraran les estratègies a utilitzar per distribuir els senyals de rellotge dins d'un sistema. L'objectiu que es tracta d'assolir és aconseguir que el senyal de rellotge arribi a tots els components del sistema amb un soroll temporal (skew o jitter) mínim. Es consideren dins d'aquesta assignatura únicament els aspectes de distribució de rellotge entre components del sistema. Es a dir, no es tractaran estratègies de distribució del rellotge a un sistema integrat, doncs aquests aspectes estan tractats amb detall a altres assignatures.

• Clock distribution within the system (off-chip):

- Tuning of the line length and the buffer capacitance
- Clock distribution tree:

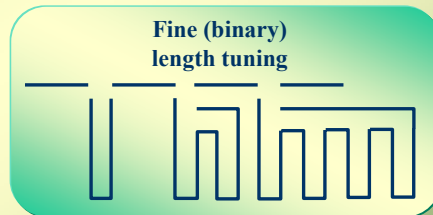
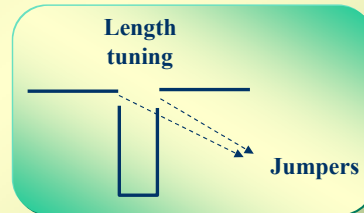


2.2. Distribution of clock signals

La distribució del senyal del rellotge dins d'un sistema és un procés relativament senzill. Els únics elements que intervindran en aquest procés seran línies i buffers. Per tant, per tal de garantir una correcta distribució del senyal de rellotge només caldrà tenir compensada la longitud de les línies i ajustades les capacitats dels buffers. A la part inferior de la imatge es mostra l'estructura típica d'un arbre de distribució de rellotge. A partir d'un oscil·lador primari el senyal de rellotge es distribuirà a tot el sistema fent servir una jerarquia de buffers (primaris, de placa i de component) i línies de transmissió (cables o pistes a una placa de circuit imprès). Les consideracions a tenir en compte per garantir una correcta distribució del senyal de rellotge consistiran en garantir que tots els buffers tinguin el mateix fan-out i que les línies de distribució associades a cada nivell jeràrquic estiguin compensades en longitud.

• Tuning/compensation of line length:

- **Wires:** Cut with the same length
- **Lines in a board:**

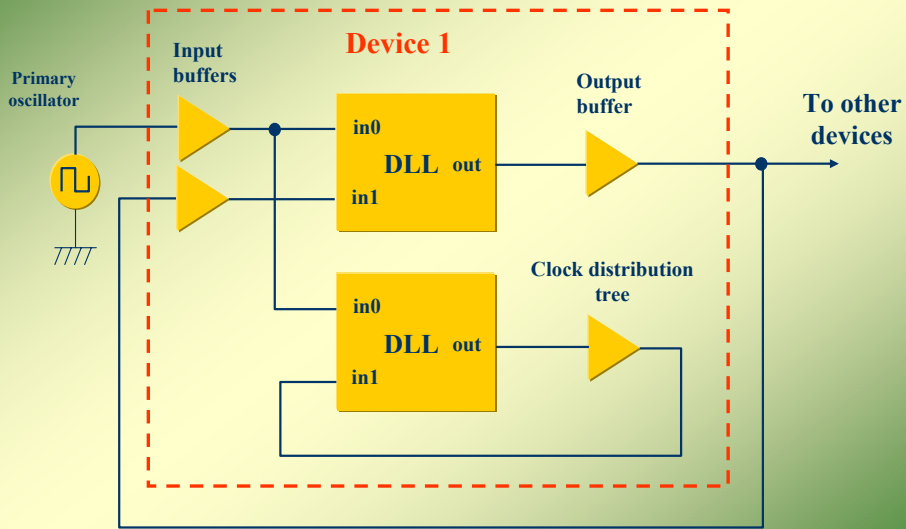


2.2. Distribution of clock signals

A la imatge es mostra com es poden ajustar o compensar les línies de distribució. En el cas que aquestes siguin cables es tractarà de garantir que durant el procés de tall dels mateixos tots els segments tenen la mateixa longitud. Pel que fa a les línies a una placa de circuit imprès una forma de compensar la longitud es utilitzar geometries en forma de trombó com la que es mostren a la part central esquerra de la imatge. En aquesta secció la línia inferior té la mateixa longitud que la superior, tot i que es fa servir per connectar components que estan a una distància menor. L'ajust de la longitud de les línies a una placa de circuit imprès es pot aconseguir inserint, fent servir jumpers, seccions de línia de longitud prefixada entre els segments de línia que ha de connectar dos components, tal i com es mostra a la part central dreta de la imatge.

A la part inferior de la imatge es mostra com es pot fer un ajust fi de la longitud de les línies utilitzant seccions de línia la longitud de les quals creix amb la potència de 2.

• Clock distribution with DLLs:

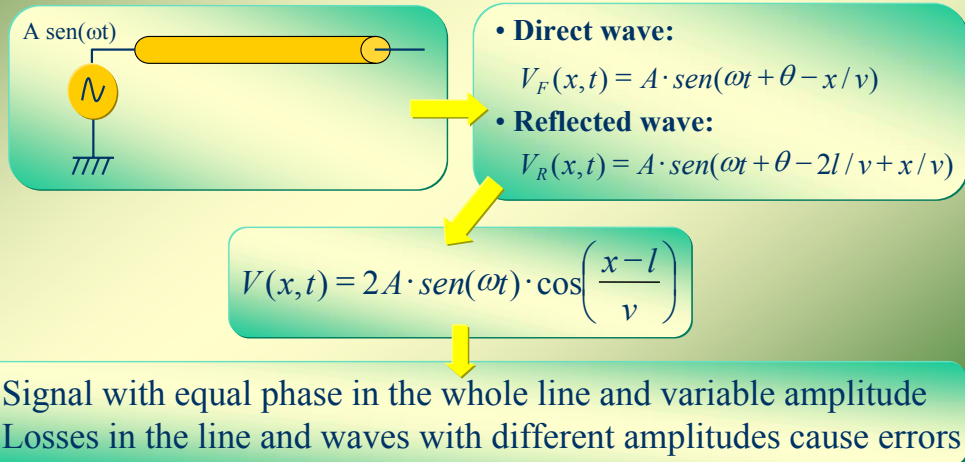


2.2. Distribution of clock signals

La distribució de senyals de rellotge dins d'un sistema es pot dur a terme de forma alternativa utilitzant DLLs, tal i com es mostra a la imatge. Aquesta alternativa és cada vegada més viable, donat que pràcticament tots els dispositius programables d'última generació inclouen aquest tipus de subsistemes. A la imatge es mostra com un senyal de rellotge procedent d'un oscil·lador primari arriba a un cert dispositiu. Dins d'aquest dispositiu, el DLL situat a la part inferior de la imatge s'encarrega de garantir que el senyal de rellotge que arriba a tots els elements de memòria del sistema tingui la mateixa fase que el senyal d'entrada. D'altra banda, el DLL situat a la part superior del dispositiu, mitjançant una realimentació externa, s'encarrega de proporcionar finalment un senyal de rellotge per a un altre dispositiu que estarà en fase amb el senyal procedent de l'oscil·lador primari. Per tal de garantir que els senyals estiguin en fase és molt important que el segment de línia utilitzat per fer aquesta realimentació tingui la mateixa longitud que el segment de línia que connecta la sortida del dispositiu amb l'entrada de rellotge dels altres dispositius.

• Clock distribution based on transmission lines:

➤ Based on stationary waves present in the line:



2.2. Distribution of clock signals

Finalment, una altra alternativa per distribuir senyals de rellotge consisteix en la utilització de línies de transmissió. Efectivament, si una línia de transmissió no està acabada al seu extrem, o si en aquest extrem està connectada una impedància reactiva, llavors apareixerà sobre la línia una ona estacionària. Suposem un cas com el que es mostra a la part central dreta de la imatge, al qual hi ha un oscil·lador sinusoidal que actua com a driver d'una línia de transmissió que no està terminada. Amb aquesta situació apareixeran sobre la línia una ona directa i una ona reflectida com la que es mostren a la part dreta de la imatge. La suma d'aquestes dues ones dona com a resultat l'ona que apareixerà a la línia, que tindrà l'expressió mostrada a la part central de la imatge.

Com es pot deduir fàcilment, aquesta ona tindrà una amplitud que dependrà de la posició a la línia, i la seva fase serà la mateixa a tots els punts de la línia (descartant evidentment aquells punts als quals la seva amplitud sigui zero). Per tant, aquesta estructura es podria utilitzar per distribuir senyals de rellotge dins d'un sistema. A la pràctica, el fet que la línia de transmissió tingui pèrdues, així com el fet que les dues ones no tindran la mateixa amplitud poden donar lloc a errors de fase.