

MOOC Réseaux Locaux

Le réseau local Ethernet

Les supports physiques

Objectifs

Cette leçon a pour but de décrire brièvement les principaux supports physiques du réseau local Ethernet.

Prérequis

Connaissances de base des réseaux.

Connaissances

Principaux problèmes et pistes de solutions pour la mise en œuvre d'un réseau local sur un support physique.

Compétences





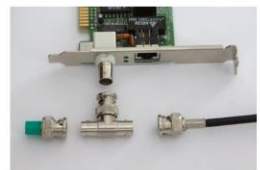

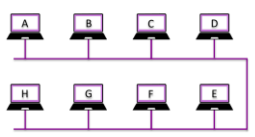


Analyser les difficultés de la transmission sur un support physique.

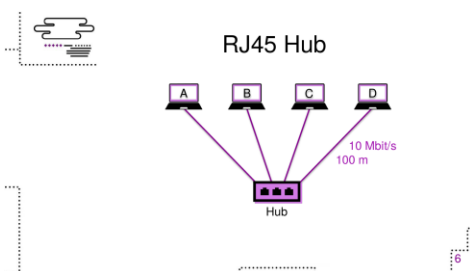
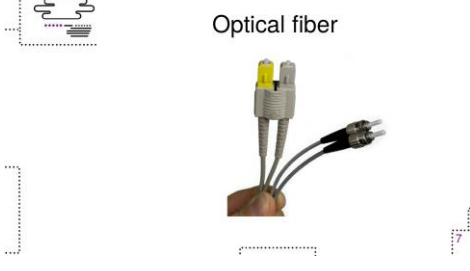
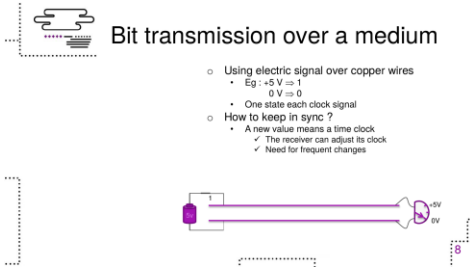
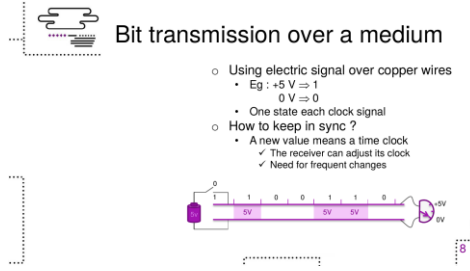
Évaluation des connaissances




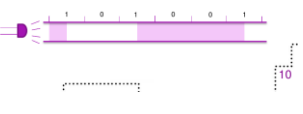



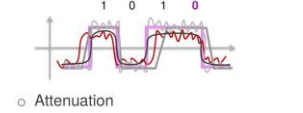
Donner les principales caractéristiques de la couche physique d'un réseau local.

Évaluation des compétences

Analyser une norme de couche physique d'un réseau local.

 <p>Ethernet <i>Ethernet physical layers</i></p> <p>Emmanuel Chaput</p>	<p>Cette leçon est consacrée à l'étude des supports physiques utilisés par Ethernet. Nous allons donc essayer de comprendre comment transmettre des 0 et des 1 sur un support physique.</p> <p>Avant de comprendre comment transmettre des 0 et des 1, bien sûr nous devons nous poser la question de quel support physique nous allons utiliser.</p>
 <p>Thin ethernet medium</p>  <p>10 Mbit/s 2.5 km (5x500 m)</p>	<p>L'un des premiers câbles avoir été largement utilisés par Ethernet dans les années 1980 est un câble coaxial doté à ses extrémités d'une prise qualifiée de prise baïonnette ou prise BNC. Ce câble, relativement rigide, permettait d'atteindre des débits de 10 mégabits par seconde sur des distances de l'ordre de 2,5 km.</p>
 <p>Legacy medium</p> 	<p>Les machines étaient connectées sur ce câble à l'aide d'équipements tels que ce que vous observez ici</p>
 <p>Bus topology</p> 	<p>et nous obtenions ainsi une topologie de bus que nous pouvons présenter de cette façon.</p>
 <p>Twisted pair with RJ45 plug</p> 	<p>Dans les années 1990, l'utilisation de ce type de câble comportant quatre paires de fils de cuivre torsadées s'est généralisée et Ethernet a donc utilisé plus massivement ce support qui présente l'avantage d'être sensiblement plus souple et surtout sensiblement moins cher, bien qu'avec une efficacité sensiblement inférieure à celle du câble que nous avons vu précédemment en particulier en termes de distance.</p> <p>Le connecteur traditionnellement utilisé est un connecteur dit RJ45.</p>

 <p>RJ45 Hub</p> <p>10 Mbit/s 100 m</p>	<p>Du fait des faibles longueurs qui peuvent être atteintes avec ce type de câble, il n'est plus question de faire de serpenter un bus de 2.5 km dans les bâtiments et de brancher les machines dessus. Du coup la topologie qui est utilisée avec les câbles paires torsadées et leur prise RJ45 est la topologie illustrée par cette figure dans laquelle un élément central, le hub, est relié à toutes les machines au travers de ces câbles.</p> <p>Nous obtenons donc ainsi une topologie physique en étoile. Attention cependant, il ne faut pas se laisser duper : la topologie électrique reste strictement la même : il s'agit une topologie de bus.</p>
 <p>Optical fiber</p>	<p>Fin des années 90, l'utilisation de la fibre optique a permis d'atteindre des débits sensiblement plus élevés y compris sur des distances plus grandes. Vous voyez ici un exemple de connecteurs associés à cette fibre optique.</p>
 <p>Bit transmission over a medium</p> <ul style="list-style-type: none"> Using electric signal over copper wires <ul style="list-style-type: none"> Eg : +5 V \Rightarrow 1 0 V \Rightarrow 0 One state each clock signal How to keep in sync ? <ul style="list-style-type: none"> A new value means a time clock The receiver can adjust its clock Need for frequent changes 	<p>Maintenant que nous savons sur quel type de support nous pouvons envoyer nos 0 et nos 1, posons-nous effectivement la question de comment transmettre ces 0 et ces 1. Un exemple très simple et très illustratif est de considérer du fil de cuivre sur lequel nous voulons transmettre des 0 et des 1 la machine émettrice va par exemple émettre une tension de plus 5 volts pour manifester un 1 et une tension nulle pour faire comprendre au récepteur qu'elle veut lui envoyer un 0.</p>
 <p>Bit transmission over a medium</p> <ul style="list-style-type: none"> Using electric signal over copper wires <ul style="list-style-type: none"> Eg : +5 V \Rightarrow 1 0 V \Rightarrow 0 One state each clock signal How to keep in sync ? <ul style="list-style-type: none"> A new value means a time clock The receiver can adjust its clock Need for frequent changes 	<p>D'aussi bonnes qualités qu'elles soient, malheureusement les horloges de l'émetteur et du récepteur ne peuvent pas être parfaitement équivalentes. On dit qu'elles vont dériver l'une par rapport à l'autre. Quelle en est la conséquence ? Le récepteur va essayer de lire un bit un moment inopportun et il va lire deux bits là où il ne fallait en lire qu'un ou il va en lire un là où il fallait en lire 2. Nous avons clairement un problème de synchronisation.</p> <p>Quelle réponse doit être apportée à ce problème de synchronisation ? La solution classique est d'asservir l'horloge du récepteur sur celle de l'émetteur : le récepteur va observer les changements d'état du support, les changements de tension dans notre exemple, et il va essayer de voir si ces changements arrivent un peu trop vite ou un peu trop lentement par rapport à son horloge. Comme il ne peut pas changer le rythme auquel les symboles arrivent, bien entendu, que va-t-il faire ? Il va changer un petit peu son horloge de façon à s'adapter à cela.</p> <p>S'il n'y a pas de changement d'état, malheureusement, le récepteur ne pourra pas asservir son horloge.</p>

 <h3>Synchronization issue</h3> <ul style="list-style-type: none"> Need for frequent changes <ul style="list-style-type: none"> Eg : «0 V then 5 V» \Rightarrow 1 «5 V then 0 V» \Rightarrow 0 Used by Ethernet (IEEE 802.3) with -2.5 V / +2.5 V <ul style="list-style-type: none"> ✓ Manchester 	<p>Une des solutions consiste par exemple à imposer un changement d'état à chaque fois que l'on transmet un bit.</p> <p>Par exemple, un bit 1 sera matérialisé par un changement d'état 0 volts 5 volts et un bit 0 par un changement d'état 5 volts 0 volts. C'est exactement ce type de codage qu'utilise Ethernet avec des tensions en l'occurrence de 2.5 volts. Ce type de codage s'appelle Manchester.</p>
 <h3>Optical fiber</h3> <ul style="list-style-type: none"> Ethernet 100BaseFX uses NRZI <ul style="list-style-type: none"> Any change \Rightarrow 1 No change \Rightarrow 0 Synchronization issue <ul style="list-style-type: none"> Long series of 0 	<p>D'autres types de codage existent, par exemple sur la fibre optique, on utilise un codage qui s'appelle NRZI pour <i>non return to zero inversed</i> (Sans Retour à Zéro Inversé).</p> <p>Le principe est relativement simple : on émet une lumière où on n'émet pas de lumière. Le 1 va être codé par un changement d'état systématique, c'est-à-dire, lumière allumée, je passe à lumière éteinte, lumière éteinte je passe à lumière allumée. Un 0 va être par l'absence de changement d'état : je reste entièrement éteint, ou je reste entièrement allumé.</p>
 <h3>Synchronization : 4B5B</h3> <ul style="list-style-type: none"> For 4 data Bits, send 5 Bits <ul style="list-style-type: none"> At least one 1 It is bloc coding Called 4B5B 31 possible blocs <ul style="list-style-type: none"> 16 configurations needed 15 configurations usable for signalling <ul style="list-style-type: none"> ✓ Start of frame, ... 	<p>Se repose ici à nouveau le problème de la synchronisation, dont la solution proposée ici est assez astucieuse. Elle consiste en fait à prendre les bits quatre par quatre et d'envoyer non pas quatre, mais cinq bits en assurant que parmi ces cinq bits l'un au moins soit non-nul. On assure ainsi qu'il n'y aura pas de séquences de plus de 4 bits nuls et que donc le récepteur pour assurer l'asservissement de son horloge.</p> <p>Vous remarquez bien sûr qu'avec cinq bits, on peut transmettre 32 combinaisons possibles. En fait, concrètement trente-et-une puisque celle comportant cinq fois le bit 0 est exclue. Or, nous avons envoyé 4 bits donc 16 configurations possibles.</p> <p>Il reste donc 15 configurations qui ne sont pas nécessaires. Elles pourront être utilisées par exemple à des fins de signalisation par exemple pour signaler le début d'une trame.</p>
 <h3>Mediums are not perfect</h3>  <ul style="list-style-type: none"> Attenuation Distortion Noise Some errors may occur 	<p>Nous savons maintenant transmettre des 0 et des 1 sur un support, malheureusement, vous vous en doutez, les supports de communication ne sont pas parfaits. Si bien que lorsqu'on va vouloir transmettre une séquence binaire, l'émetteur va construire un signal électrique ou optique qui va partir de façon très propre très carrée, mais qui va malheureusement subir les aléas du support. Il va par exemple subir de l'atténuation si bien qu'il va arriver sur le récepteur sensiblement moins puissant qu'il n'est parti. Il va également subir de la distorsion si bien qu'il va arriver sur le récepteur déformé par rapport à sa forme initiale. Il va également subir du bruit de l'environnement électromagnétique par exemple si bien qu'il va arriver très perturbé sur le récepteur.</p>

	<p>Évidemment, vous vous en doutez, tous ces phénomènes se conjuguent si bien que le joli signal émis par l'émetteur va se retrouver assez largement dégradé sur le récepteur qui malheureusement peut commettre des erreurs d'interprétation et voir des 0 là où il y a des 1 ou le contraire.</p>
<div data-bbox="119 436 215 504"></div> <div data-bbox="327 459 486 492"><p>Error detection</p></div> <div data-bbox="311 515 534 571"></div> <div data-bbox="287 582 566 672"><ul style="list-style-type: none">o Delivering erroneous bits is an issueo Error detection techniques introduce redundancy bitso Some errors can be detected</div> <div data-bbox="558 660 590 705"><p>13</p></div>	<p>Il va falloir trouver des contre-mesures à ce problème. En effet, la délivrance de bits erronés n'est absolument pas acceptable. Il faut être capable de s'apercevoir qu'il y a des problèmes. Une solution consiste à introduire ce que l'on appelle des bits de redondance. Prenons un exemple simple : au bout de 4 bits de données, nous ajoutons un bit qui est calculé comme étant la somme des 4 bits précédents. On appelle cela un bit de parité et nous transmettons ces cinq bits. Cette fois-ci, s'il y a une erreur, le récepteur va pouvoir s'en rendre compte, car s'il calcule la somme des quatre premiers bits, il va bien s'apercevoir qu'elle n'est pas égale à la somme que lui a envoyée l'émetteur. Il sera ainsi capable de s'apercevoir qu'il y a une erreur et donc éventuellement de réagir, d'abandonner les données par exemple. Attention, bien sûr, malheureusement ces techniques ne sont pas infaillibles !</p>
<div data-bbox="119 985 215 1052"></div> <div data-bbox="343 996 446 1030"><p>Summary</p></div> <div data-bbox="287 1064 518 1220"><ul style="list-style-type: none">o Different physical media<ul style="list-style-type: none">• Accomodate physical propertieso Coding schemeo Medium imperfection<ul style="list-style-type: none">• Error detection</div> <div data-bbox="470 1041 590 1108"></div> <div data-bbox="462 1164 566 1209"></div> <div data-bbox="566 1209 598 1254"><p>14</p></div>	<p>Que faut-il retenir de cette leçon ? Au fil du temps diverses technologies ont été utilisés pour transmettre des trames Ethernet des techniques de codage sont développés à chaque nouvelle technologie afin de s'adapter à ses caractéristiques propres et que malheureusement un élément commun à toutes ces technologies et qu'elles sont imparfaites et qu'il va donc falloir introduire des techniques de détection d'erreurs.</p>