INTRODUCTION AU BUS I²C

I. HISTORIQUE ET PRESENTATION

Devant le besoin d'un bus **"universel"**, simple, peu coûteux, peu encombrant et facile à mettre en œuvre, Phillips crée, pour ses applications *produits bruns*, au début des années 80, le bus **I²C**.

Les caractéristiques d'un tel bus, faisant partie des *L.A.N.* (réseaux locaux) doivent permettre :

- ✍ d'utiliser toutes les technologies actuelles (TTL, NMOS, PMOS, CMOS, …),
- ✍ d'être véhiculer par tout type de supports (câbles simples ou blindés, coaxiaux, fils en nappe, fibre optique, liaisons infrarouges ou hertziennes, …),
- ✍ un coût d'utilisation le plus faible possible (et donc de fabrication),
- ✍ une implantation du plus faible encombrement (réduction du nombre de composants, du nombre de broches par composant, …),
- ✍ une rapidité de transmission suffisante pour ne pas ralentir les matériels plus performants du moment,
- ✍ une non-susceptibilité et non-pollution (électronique) de l'environnement (C.E.M. oblige),
- ✍ ……

De fait, Phillips aboutit à un standard qui est aujourd'hui reconnu et utilisé dans de nombreux domaines (T.V., radio, audio, téléphonie, domotique, automobile, …), après avoir vu ses caractéristiques modifiées en 92 pour suivre l'évolution technologique (une nouvelle évolution est annoncée). Il est à noter que dans les domaines précités, des bus spécifiques ont vu le jour.

Phillips est donc propriétaire de ce standard, et tout fondeur (concepteur) de composants I²C doit posséder une licence acquise auprès de Phillips.
Il a alors le droit d'arborer ce logo. ⇨

Le bus I²C est un bus série sur 3 fils :

- ✍ Un fil bidirectionnel de transmission de données: *SDA* (**S**erial **Da**ta).
- ✍ Un fil de cadencement des transferts : *SCL* (**S**erial **CL**ock).
- ✍ La masse.

II. CARACTERISTIQUES STATIQUES

a) <u>Niveaux de tension et de courant</u> :

Du fait de l'universalité de ce bus envers les technologies, il a fallu trouver des niveaux de tension correspondant aux niveaux logiques acceptés par tous.

Ainsi :

✍ pour les entrées :

- ? un niveau bas est fixé à 0,3?VDD,
- ? un niveau haut à 0,7?VDD.

Ce qui donne, pour une alimentation de 5V, $V_{il\ max}$ = 1,5V et $V_{ih\ min}$ = 3,5V.

✍ pour les sorties :

- ? $V_{ol\ max}$ = 0,4V, pas de valeur spécifique pour $V_{oh\ min}$ car les sorties des 2 signaux sont de type collecteur (drain) ouvert, $V_{oh\ min}$ dépend de la tension d'alimentation.
- ? $I_{ol\ max}$ = 3mA max.

b) Résistances de tirage et de protection :

Comme toute sortie à collecteur ouvert, il est indispensable de prévoir des résistances de tirage. La mise en parallèle de l'ensemble des résistances de rappel ne doit pas être inférieure à 1,7k? .

On prévoit également une résistance de protection contre les parasites, d'une valeur maximum de quelques centaines d'ohms, et montée entre l'entrée-sortie du composant et les 2 lignes.

Pour la détermination de ces composants, voir les abaques en **ANNEXE** de ce document.

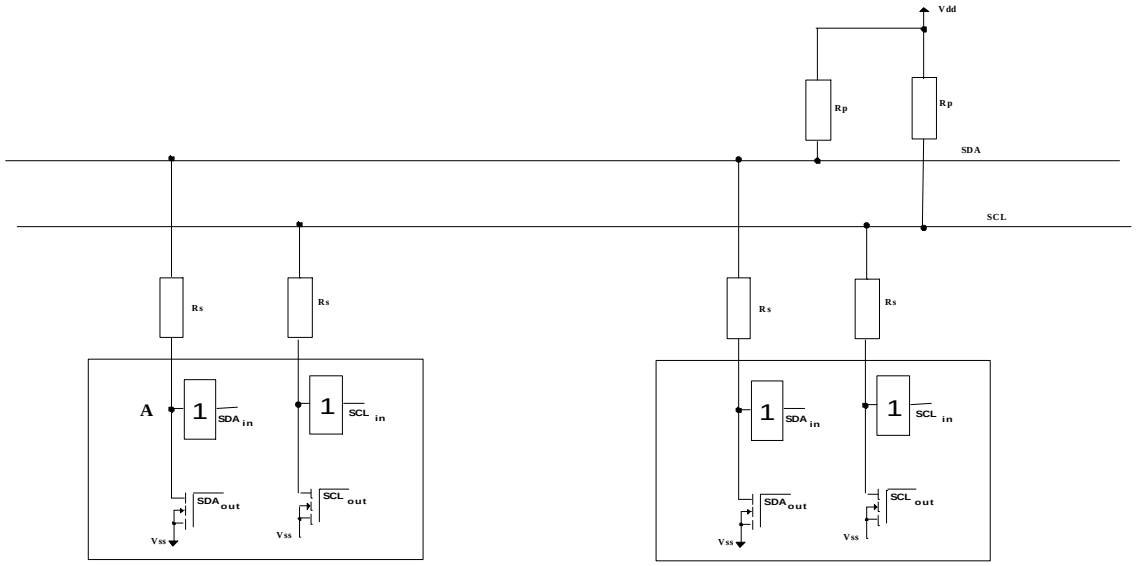


Figure 1 : Schéma des liaisons avec le bus et étages d'entrées et de sorties

c) Capacité des lignes :

Pour chacune des 2 lignes, on parle de chargeabilité (on ne peut parler d'entrance ou de sortance pour une ligne bidirectionnelle). Il s'agit donc du nombre maximum de circuits connectables sur chaque ligne.

Le standard fixe à 20pF maximum la capacité de chaque entrée-sortie et une liaison filaire à une capacité de 100pF/m.

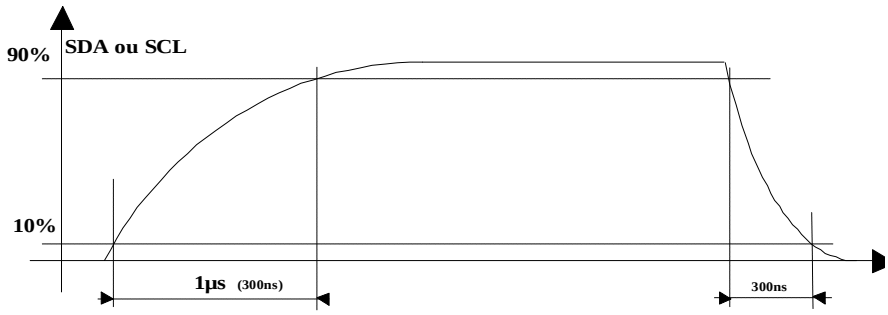
La somme de ces capacités ne doit pas excéder 400pF d'après la première version du standard puis 200pF par la suite.

III. CARACTERISTIQUES DYNAMIQUES

a) Temps de montée et de descente :

Le temps de descente ne pose pas réellement de problème car il est déterminé par le temps de décharge des capacités équivalentes au travers du transistor de sortie, lorsqu'il se sature (au max. 300ns).

Le temps de montée maximum, dans la première version du standard, est de 1µs et de 300ns dans la nouvelle.



C'est à l'aide de ce critère que l'on détermine la valeur de la résistance de tirage.

Remarque : Le temps de montée correspond à environ 2,2?.

b) Fréquence du signal de cadencement **SCL** *:*
100kHz à l'origine, portée à 400kHz par la suite.

IV. FORMAT D'UN ECHANGE

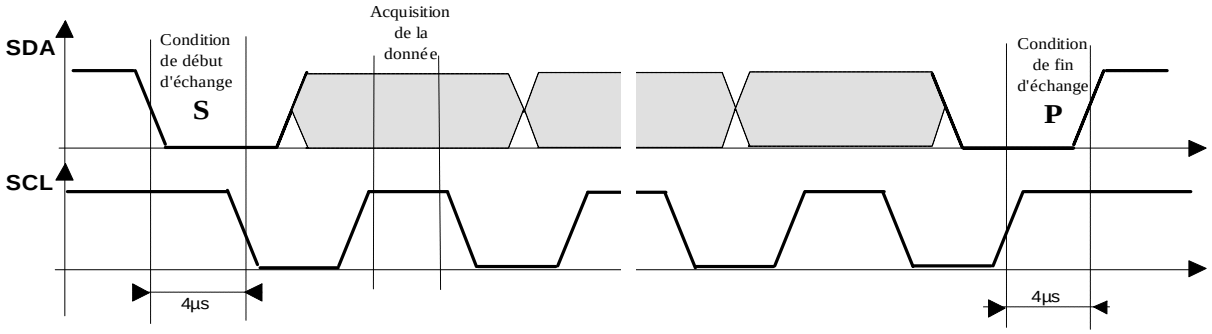
a) Indication de début et de fin d'échange :

Lorsqu'un échange doit se faire, l'émetteur génère une indication de début d'échange qui est matérialisée sur les 2 lignes, alors au repos à 1, par le positionnement de **SDA** à 0 puis 4µs (0,6µs*) plus tard, par **SCL** qui génère alors le signal de cadencement.

Les données, séries, sont alors émises sur **SDA**. Elles changent d'état lorsque **SCL** est à l'état bas et sont prises en compte pendant l'état haut de **SCL** par le récepteur.

Un échange se termine lorsque l'émetteur cesse de générer l'horloge (retour à l'état haut de **SCL** et 4µs (0,6µs*) plus tard de **SDA**).

**Les informations entre parenthèses concernent la nouvelle version du standard.*



Un nouvel échange peut avoir lieu 4,7µs après une condition de fin d'échange.

b) Identification des composants :

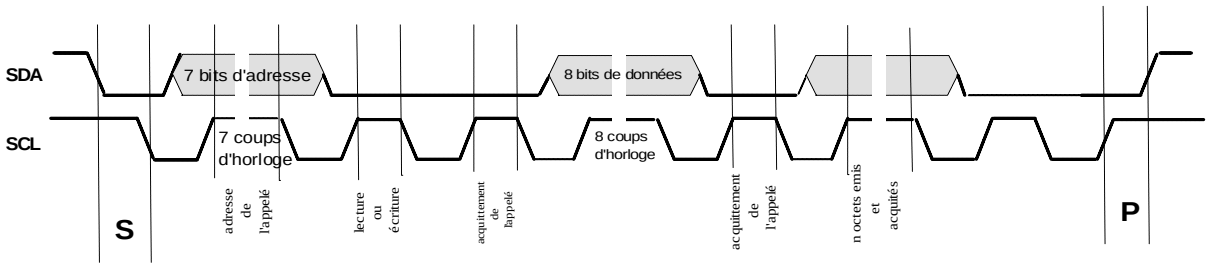
Chaque composant I²C peut être appelé par un autre composant I²C. Pour simplifier l'identification, chaque famille a une adresse spécifique (dont une partie peut être ajustée matériellement ou logiciellement). Le format de l'adressage se fait sur 7 (10*) bits.

EXEMPLE D'ADRESSES DE BASE :

Composants	Adresses	Composants	Adresses
Synthétiseur vocal	$ 10	Horloge calendrier	$ 68
Mémoire de 128 à 2048 octets	$ 50	Driver d'afficheur LCD	$ 3C
Quadruple conv. A/D et D/A	$ 48	Décodeur T.V. multi standard	$ 45

La première donnée échangée sur la ligne **SDA** est donc l'adresse de l'appelé suivie du sens de l'échange (écriture ou lecture ($R/\overline{W}$)). L'appelé, dès qu'il s'est reconnu, l'indique à l'appelant par un Acquittement (Acknowledge) en mettant à 0 la ligne **SDA**.

Puis, en fonction du sens de transfert des données, le composant qui reçoit les données acquitte chaque octet.



V. SYNCHRONISATION ET GESTION DES CONFLITS

Nous l'avons vu, sur un même bus I²C, plusieurs composants sont reliés.
Comment cela se passe-t-il lorsque :

✍ *un composant n'a pas "compris" une donnée :*

Le composant en question n'acquitte pas la donnée (il laisse à 1 la ligne **SDA** au moment de l'acquittement). L'émetteur, n'ayant pas d'accusé de réception, ré-émet la donnée.

✍ *2 composants veulent être émetteurs en même temps :*

Sur la figure 1 page 2, on constate que entrée et sortie utilisent le même fils. Le composant peut donc "s'écouter". Lorsqu'il génère un niveau logique, il vérifie sa présence sur la ligne.

Si 2 émetteurs se connectent en même temps sur la ligne, le premier qui voudra générer un niveau logique 0 alors que l'autre émetteur désire un 1, verra alors sur la ligne un 0 (collecteur ouvert oblige). L'autre émetteur constatera alors un conflit, libérera la ligne et se mettra en écoute (il peut être le composant appelé !).

✍ *un composant est plus lent que l'autre :*

Qu'il soit émetteur ou récepteur, tant qu'il n'a pas fini une opération, un composant I²C maintient la ligne **SCL** à 0, demandant à son interlocuteur "d'attendre".

VI. GESTION DES "OPERATIONS I²C"

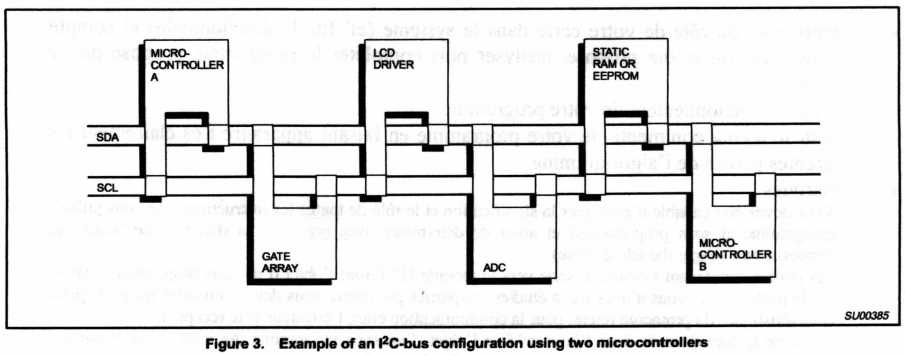


Figure 3. Example of an I²C-bus configuration using two microcontrollers

Toute structure comportant un bus I²C est de type structure microprogrammée. Elle comporte donc un microcontrôleur qui joue le rôle de *maître* alors que les autres composants sont considérés comme *esclaves* (il arrive parfois, dans les réseaux évolués que les rôles soient redistribués).

Il est à noter que bon nombre de microcontrôleurs spécialisés (dotés d'entrées-sorties spécialisées gérant directement les niveaux et le protocole I²C) existent sur le marché (ex : 80C51, 80C552, …).

Dans de petites structures, il est par ailleurs possible de générer directement les signaux à l'aide des entrées-sorties parallèles des microcontrôleurs standards et de concevoir les routines logicielles d'échanges avec le bus I²C.

VII. CARACTERISTIQUES SPECIFIQUES AUX DIVERS COMPOSANTS

Au paragraphe IV, a été abordée la procédure générale d'un échange. Chaque composant nécessite de plus, pour son fonctionnement, des informations supplémentaires qui sont passées dans le champ de données.

Intéressons-nous à la mémoire série **M24C02** dont la documentation constructeur est fournie.

Comme toute mémoire, les mémoires I²C nécessitent, pour stocker ou fournir des données, de connaître le lieu de localisation (adresse).

Examinons les phases de lecture puis d'écriture.

✍ **LECTURE**

Cette mémoire permet plusieurs modes d'accès aux données stockées. Il est possible d'aller lire une donnée unique parfaitement localisée, on parle alors d'accès ***aléatoire***, ou de lire un flot de données consécutives, il s'agit alors d'accès ***séquentiel***.

a) Cycle de lecture aléatoire :

On désire donc lire un octet unique.

La première étape consiste donc à établir une "communication" en positionnant **SDA** et **SCL** suivant la procédure de *début d'échange* (voir paragraphe IV) repérée, dans la représentation de **SDA** ci-dessous par un **S**.

Puis :

✍ Il précise l'adresse du composant (notre mémoire).

✍ Il indique le type d'échange : une écriture ($R/\overline{W} = 0$) pour que la mémoire "écoute la ligne" et sache ainsi que la donnée à suivre la concerne.

✍ La mémoire s'étant reconnue, "répond" par un *acquittement*.

✍ Le microcontrôleur précise ensuite le lieu (adresse) ou se trouve la donnée qu'il désire lire. Cette information est mémorisée dans la mémoire par un *pointeur*. Elle confirme au microcontrôleur cette mémorisation par un acquittement.

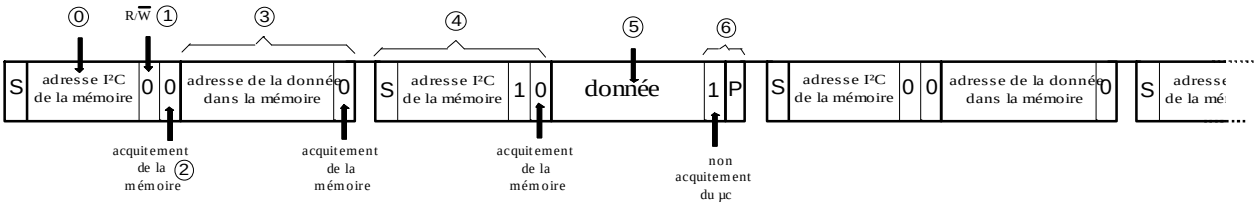
Suit maintenant la lecture de la donnée :

✍ Le microcontrôleur indique à la mémoire qu'il va la lire ($R/\overline{W} = 1$), elle acquitte cette information.

✍ Elle émet alors sur la ligne le contenu de la case mémoire dont l'adresse a été précisée au point ✍.

✍ Le microcontrôleur indique, après avoir exploité cette donnée, que l'échange est terminé en "non-acquittant" cette donnée puis en générant la procédure électrique de *fin d'échange* (**P** sur le dessin).

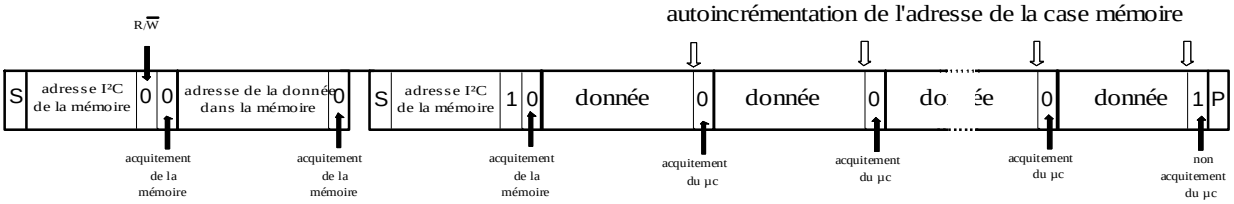
Un nouvel échange peut alors débuter.



b) Cycle de lecture séquentielle :

On désire, dans ce cas, lire une page.

Les étapes ✍ à ✍ sont identiques. A la fin de la 1ère lecture, le pointeur de case mémoire exploitée est incrémenté pour pointer la case suivante. Dès que le microcontrôleur a acquitté cette donnée, la suivante est émise. L'échange se termine lorsque le microcontrôleur le décide, en générant les niveaux de l'étape ✍.



✍ ECRITURE

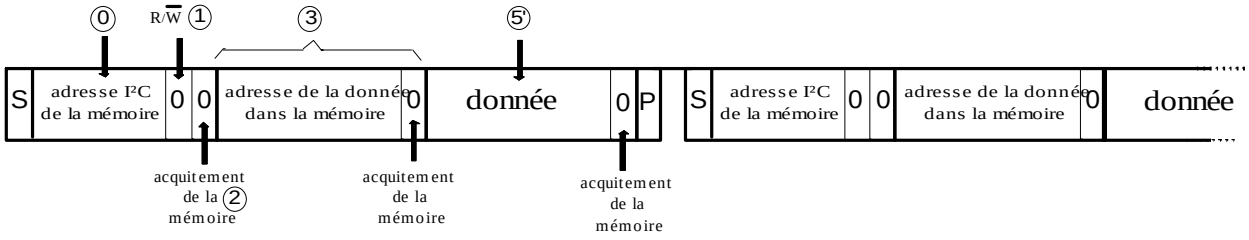
Il est également possible d'avoir des accès aléatoires ou séquentiels.

a) Cycle d'écriture aléatoire :

Pas de changement sur les étapes ✍ à ✍. Du fait qu'un cycle d'écriture est en cours, il n'est pas ici nécessaire de reproduire l'étape ✍. L'étape 5' consiste donc, pour le microcontrôleur, à présenter sur la ligne la donnée à mémoriser.

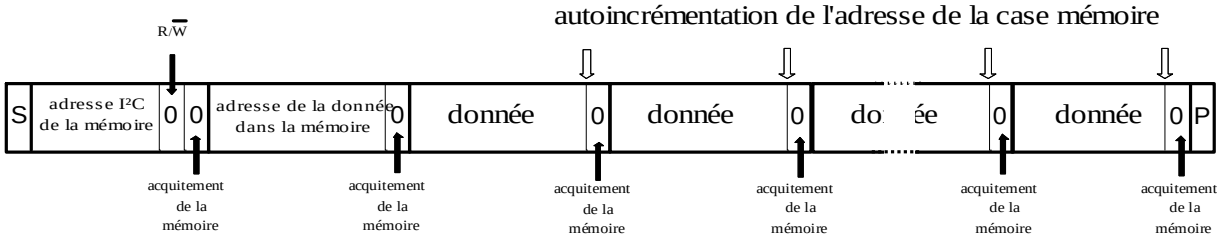
Du fait que la mémoire **M24C02** est une mémoire EEPROM, le temps d'écriture est beaucoup plus long que sur une RAM. La mémorisation d'une donnée peut prendre jusqu'à **10ms**. Durant toute cette période, la mémoire est inaccessible. Une fois la mémorisation effectuée, la mémoire l'indique par un acquittement.

Le microcontrôleur termine alors l'échange (P).



b) Cycle d'écriture séquentielle :

Dans ce mode, le principe précédent est respecté et, tout comme pour la lecture séquentielle, à l'issue de la première écriture, il y a auto-incrémentation de l'adresse.



VIII. ANNEXE

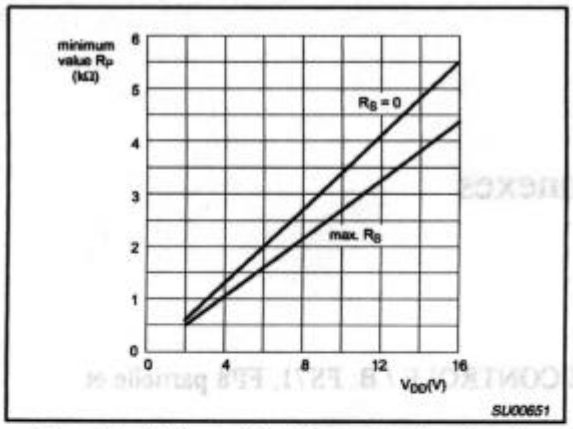


Figure 25. Minimum value of R_P as a function of supply voltage with the value of R_S as a parameter

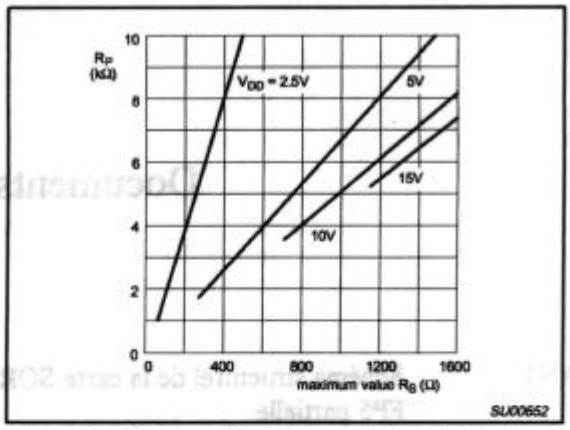


Figure 26. Maximum value of R_S as a function of the value of R_P with supply voltage as a parameter

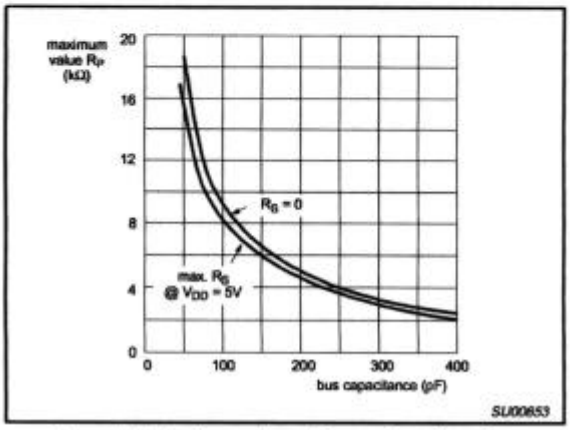


Figure 27. Maximum value of R_P as a function of bus capacitance for a standard-mode I²C-bus

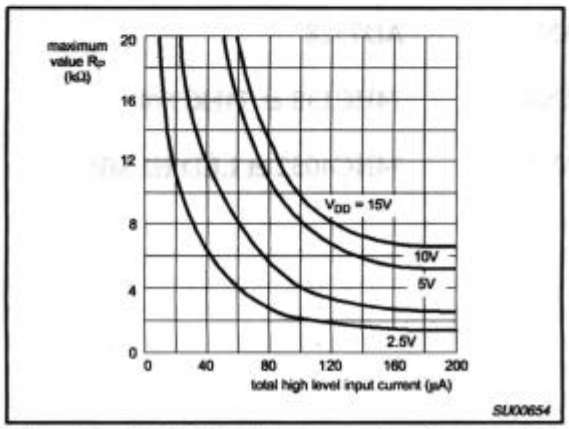


Figure 28. Total HIGH level input current as a function of the maximum value of R_P with supply voltage as a parameter