



PULSAR

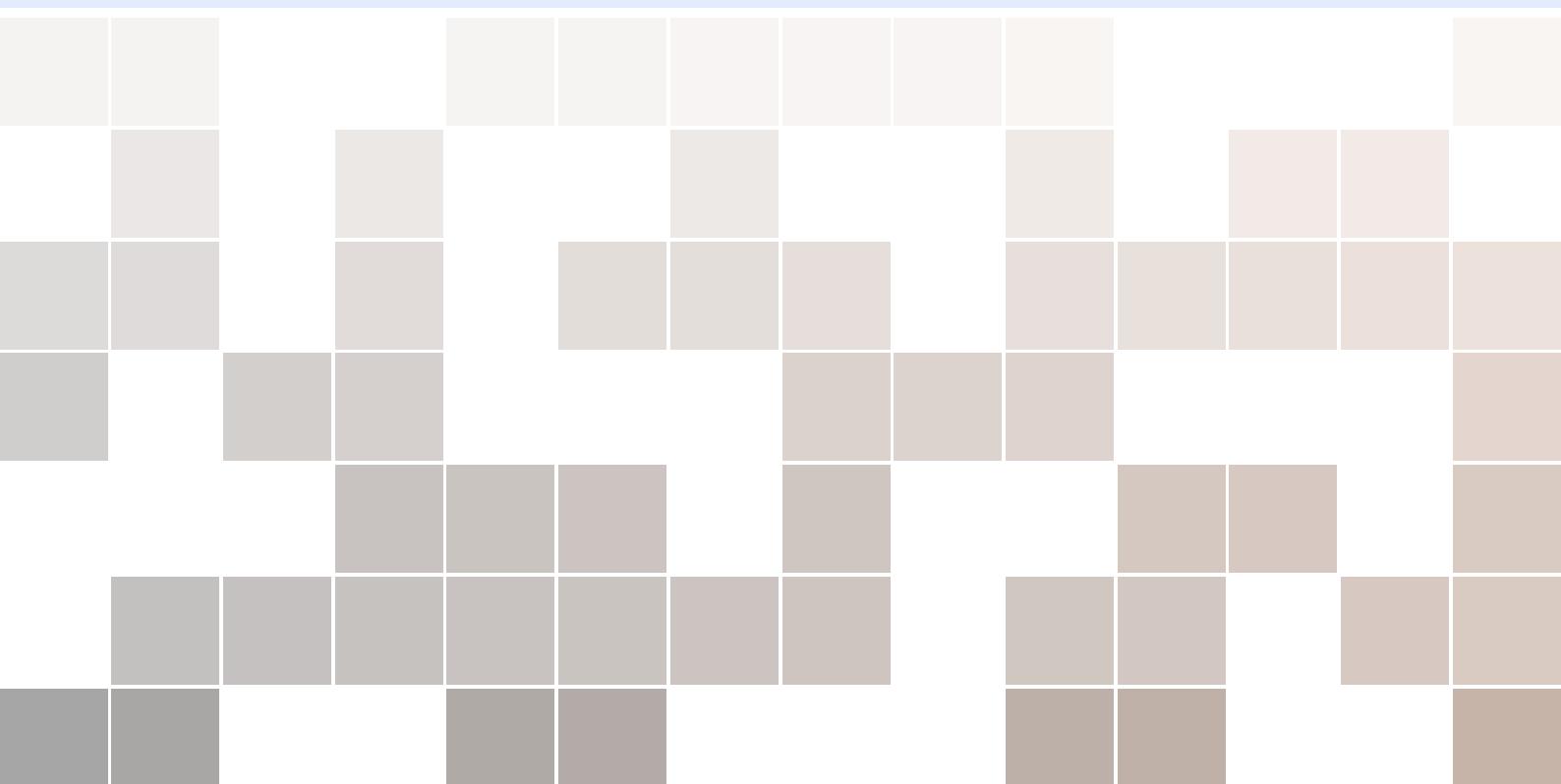
A Hybrid Digital-Analog Synthesizer for 21st Century Electronic Music

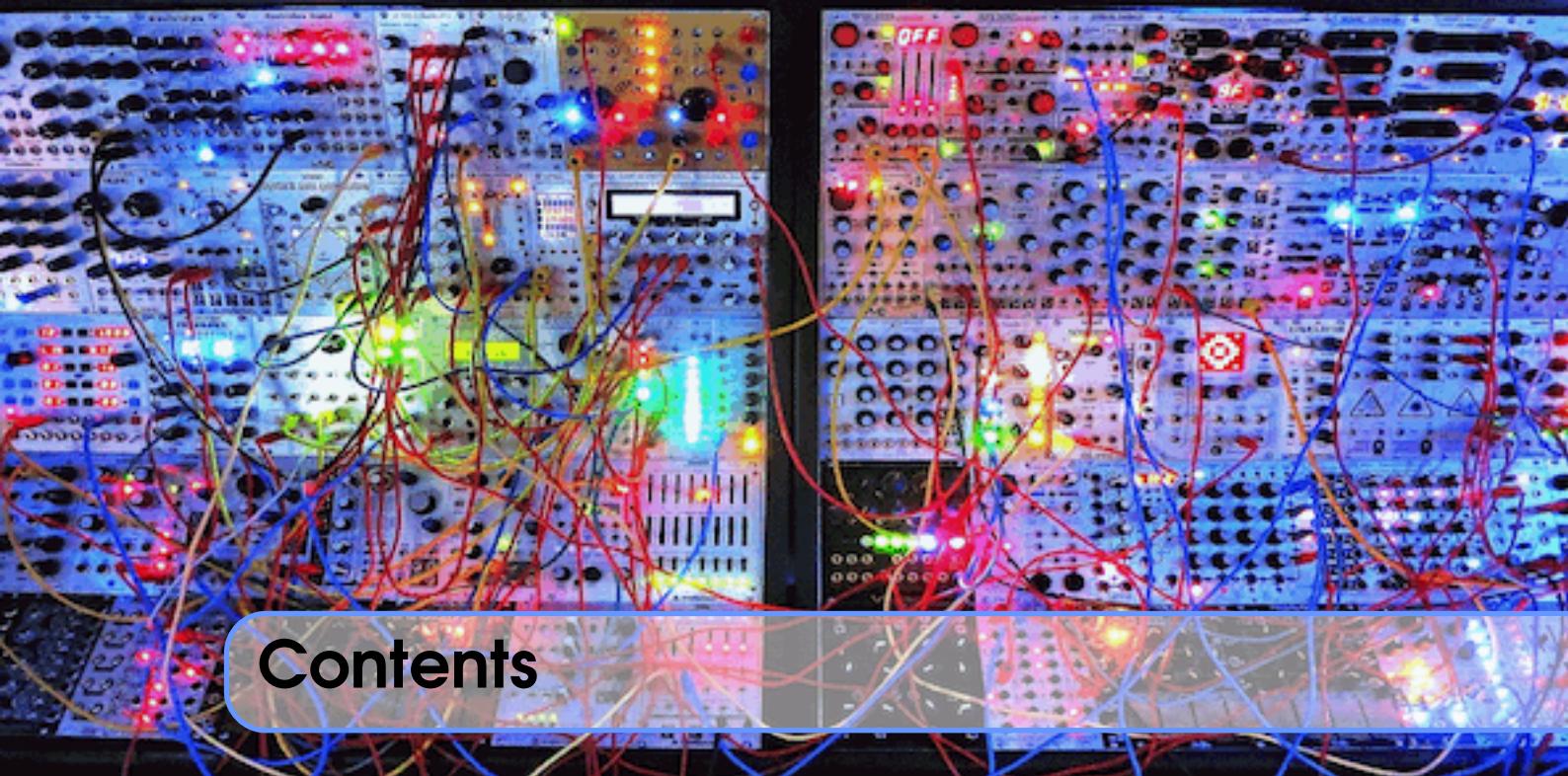
P.E. Barthelemy, V. Clabaux, C. Colart, A. Coniasse, F. Dos Santos Simao, I. El Fatihi, M. Lighoudi, N. Meerun,

A. Moutawakil, H. Rivière, A. Rozière, A. Soler — Encadrement pédagogique et technique S. Reynal

ENSEA

Septembre 2017 - Avril 2018





Contents

I

Présentation du projet par l'encadrant

1	Contexte et cahier des charges	9
1.1	Contexte historique	9
1.2	Pourquoi une approche hybride analogique-numérique ?	10
1.3	Cahier des charges initial du projet (Septembre 2017)	10
2	Architecture générale de l'instrument	13
2.1	VCO	13
2.2	VCF	15
2.3	VCA	15
2.4	Générateurs d'enveloppe ADSR	16
2.5	LFO et modulations	16
2.6	MIDI	17
2.7	Interface Homme-Machine	17
2.8	Module d'effet analogique	17
2.9	Module de distorsion à tubes	18

Modules réalisés par les étudiants en 2017-2018

3	Wavetable par N. Meerun et A. Moutawakil	21
3.1	Création des tables d'onde (N. Meerun)	21
3.2	Le rebouclage en C	22
3.3	Génération des ondes numériques "Wavetable", par A. Moutawakil 23	
3.3.1	Choix de la forme d'onde	23
3.3.2	Choix de la note	24
3.3.3	Écriture de la forme d'onde sur le CNA	24
3.3.4	Code commenté	24
4	VCO analogiques par P.E. Barthélémy	29
4.1	Généralités	29
4.2	Étude théorique de l'amplificateur à transconductance	29
4.2.1	Principe	29
4.2.2	Montages de base	30
4.3	Étude du montage effectué : LM13700	33
4.4	Montage avec mélangeur	34
5	VCF par A. Rozière/F. Dos Santos	39
5.1	Généralités	39
5.2	Caractérisation des filtres	39
5.3	Conclusions sur l'implémentation des VCF	40
6	VCA par V. Clabaux	43
7	Guide d'utilisation des cartes prototypes	47
7.1	Carte analogique (V. Clabaux)	47
7.2	Carte convertisseurs (S. Reynal)	48
8	Générateurs ADSR par C. Colart	51
8.1	Principe	51
8.2	Génération des signaux numériques par la carte	52
9	Face avant par I. El Fatihi et M. Lighoudi	53
9.1	Principe	53
9.2	Pourquoi des Leds Bars ?	56
9.3	Qu'est ce qu'un expander MCP23017?	56

10	Interface MIDI par M. Lighoudi	57
10.1	Principe	57
10.2	L'interface Midi du PULSAR	58
11	Module distortion (A. Coniasse, H. Rivière)	63
11.1	Cahier des charges	64
11.2	Architecture du préamplificateur	65
11.3	Ecart au cahier des charges et améliorations du prototype	69
12	Module delay par A. Moutawakil et A. Soler	77
12.1	Fonctionnement Général	78
12.1.1	Ergonomie et utilisation	80
12.1.2	Cahier des charges	81
12.2	Fonctionnement du circuit	81
12.2.1	Schéma global et fonctionnement	81
12.2.2	Filtre amont - Passe bas Anti-aliasing	81
12.2.3	Filtre aval - De reconstruction du signal	82
12.2.4	Réinjection	82
12.2.5	Circuit de génération des horloges	82
12.3	Réalisation	82
12.3.1	Schéma Eagle - S. Reynal	82
12.3.2	Programmation des horloges	82
A	La Java Native Interface	81
B	Code C et Java produit en 2017-2018 (RPi3)	83
B.1	Code création des tables d'onde (N. Meerun)	83
C	Schémas et datasheets	87

Présentation du projet par l'encadrant

1 Contexte et cahier des charges . 9

- 1.1 Contexte historique
- 1.2 Pourquoi une approche hybride analogique-numérique ?
- 1.3 Cahier des charges initial du projet (Septembre 2017)

2 Architecture générale de l'instrument 13

- 2.1 VCO
- 2.2 VCF
- 2.3 VCA
- 2.4 Générateurs d'enveloppe ADSR
- 2.5 LFO et modulations
- 2.6 MIDI
- 2.7 Interface Homme-Machine
- 2.8 Module d'effet analogique
- 2.9 Module de distorsion à tubes

1. Contexte et cahier des charges

1.1 Contexte historique

L'histoire de la synthèse sonore "électrique" démarre d'une certaine façon avec les ondes martenots et les premiers piano électriques Rhodes après-guerre. Le cœur de ces dispositifs est constitué d'oscillateurs électriques à base de condensateurs, résistances et inductances. Le son est une oscillation de pression qui se propage de proche en proche dans l'air : munis d'un amplificateur et d'un haut-parleur, ces instruments étaient donc capable d'émettre un son, les synthétiseurs étaient nés.

Au cours des années 60 puis 70, la synthèse sonore évolue sous l'impulsion de grandes marques comme Moog ou Korg, et s'éloigne du modèle initial des piano électriques et des orgues Hammond pour générer des sons nouveaux, artificiels, qui n'ont parfois qu'une lointaine parenté avec les instruments acoustiques de l'orchestre qu'ils étaient destinés à imiter. Plusieurs schémas de synthèse voient le jour au cours de cette période, mais c'est la synthèse soustractive qui emporte l'adhésion des musiciens et des fabricants : ici, un ou plusieurs oscillateurs produisent une forme d'onde riche en harmonique (carré, triangulaire, etc), et un filtre passe-bas est chargé de couper plus ou moins fortement la partie haute du spectre au cours du temps, imitant en cela le mécanisme d'atténuation des hautes fréquences à l'œuvre dans les instruments acoustiques. C'est ce schéma de synthèse que nous retrouvons dans les célèbres Minimoog Voyager, Korg MS20 ou Prophet 2000 dont les sonorités ont constitué l'empreinte sonore de centaines d'albums des 70's et 80's, de Pink Floyd à Kraftwerk ou Depeche Mode.

Au cours des années 80, d'autres schémas émergent, rendus possible par l'intégration poussée des fonctions analogiques (amplification, filtrage) sur un seul chip. Ainsi, la synthèse FM se distingue comme le nouveau paradigme permettant de générer des sons totalement nouveaux, en particulier les sons anharmoniques produits par les cloches, les carillons, xylophones, ... Le DX7 de Yamaha, avec sa synthèse FM à 6 oscillateurs et sa large polyphonie (il est possible de jouer des accords de près de 8 notes simultanément) est

l'emblème de cette technique de synthèse, et ses sonorités sont indéniablement associées au "son eighties", de The Cure à Tears for Fears en passant par Michael Jackson, Prince ou Madonna.

Avec l'arrivée d'architectures numériques performantes dans les années 90, c'est la synthèse numérique qui remplace progressivement les technologies analogiques : un microprocesseur génère une forme d'onde plus ou moins complexe, lui applique un filtre dont la coupure dépend du temps, module son amplitude, etc., l'approche purement numérique autorisant des matrices de modulations très variées. C'est aussi au cours des 90's que les samplers prennent leur envol, grâce à l'augmentation de la capacité des mémoires. Ici, les formes d'onde correspondent à des échantillons stockés en mémoire et non à des formes d'ondes calculées en temps réel. On retrouve notamment cette approche dans la série S2000/S5000 et dans les MPC d'Akaï, largement utilisées dans le hip-hop des 90's/2000 chez des artistes comme Beastie Boys, Run DMC, Eminem, Missy Elliott ou Public Enemy.

1.2 Pourquoi une approche hybride analogique-numérique ?

Nous assistons désormais, et ce depuis le tournant des années 2010, à l'émergence d'une approche hybride mélangeant analogique et numérique, sous l'impulsion de marques prestigieuses comme Dave Smith Instruments à San Francisco, ou Arturia à Paris. L'électronique analogique retrouve en effet depuis quelques années un regain d'intérêt dans le domaine de la synthèse sonore, des racks d'effet et des consoles de mixage par sa capacité à "ré-humaniser" la chaîne de traitement sonore : l'idée est d'enrichir le rendu sonore en insérant des composants analogiques à des endroits judicieux dans la chaîne de traitement audio afin d'exploiter leurs non-linéarités, la grande variabilité de leurs caractéristiques, leur dépendance naturelle à la température, autant de propriétés qui restent lourdes et coûteuses à modéliser dans des systèmes purement numériques "temps réel".

L'intérêt de l'hybridation est également de maintenir un "signal path" (le chemin que parcourt le signal de la source jusqu'à la sortie jack du synthétiseur) qui soit essentiellement analogique, mais de contrôler ces composants via des micro-contrôleurs numériques. On peut espérer, avec cette approche, contourner le principal défaut des systèmes de synthèse purement analogiques : la nécessité de (re)calibrer régulièrement les composants (les fréquences des oscillateurs en particulier, qui influencent directement la justesse de l'instrument).

A titre d'exemple concret, un oscillateur basé sur le circuit CEM3340 de la marque Curtis, produit une onde triangulaire dont la fréquence est contrôlée par la tension présente sur la broche 15. La justesse de l'instrument dépend donc de la capacité à produire une tension précise. Alors qu'une telle tension était générée par un diviseur de tension dans les machines des 80's, aujourd'hui elle peut être produite par un convertisseur numérique-analogique et un micro-contrôleur : elle est donc plus stable dans le temps, plus précise, et le calibrage tension-fréquence (e.g., la connaissance de la tension à générer sur la broche 15 pour produire la note "La4") peut être directement réalisé par le microcontrôleur.

La même approche vaut pour les racks d'effet et les préamplificateurs, dont les caractéristiques peuvent désormais être ajustées par microcontrôleur, tout en maintenant un signal path 100% analogique de bout en bout de la chaîne.

1.3 Cahier des charges initial du projet (Septembre 2017)

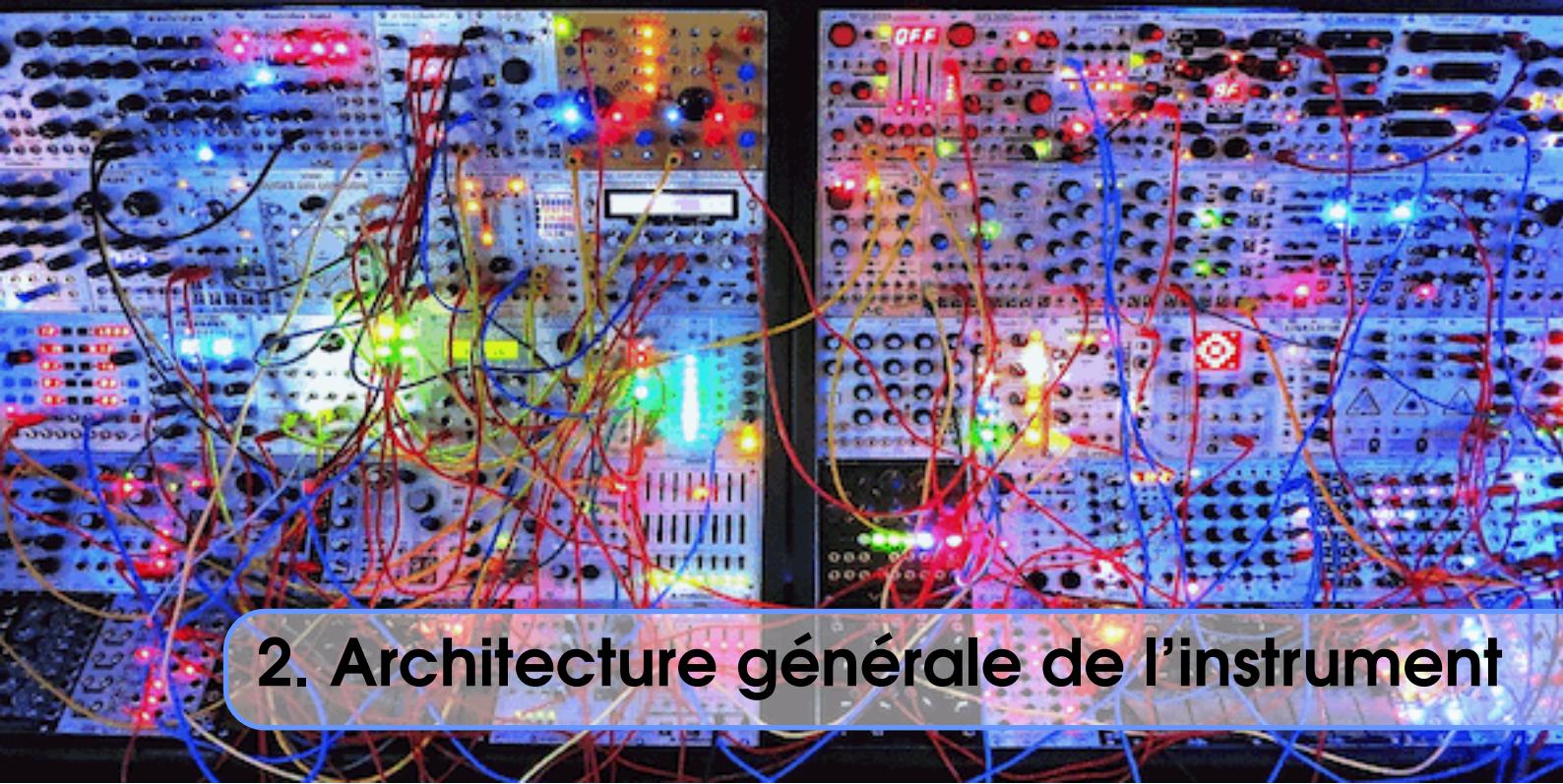
L'objectif était de concevoir un instrument sous forme de modules relativement indépendants (VCO, VCF, VCA, génération d'enveloppe, module d'effets en sortie, préamplificateur à tube) que l'on intégrerait et connecterait lors de l'assemblage dans le boîtier final. Il s'agit donc plus exactement d'un synthétiseur semi-modulaire, donc les modules sont pré-connectés. L'intérêt est de permettre une conception en parallèle distribuée sur plusieurs étudiants, mais aussi de s'autoriser à re-définir l'architecture en cours de route sans avoir besoin de redessiner schémas électroniques et PCB.

Le caractère hybride de la machine est due à la coexistence :

- d'un signal path analogique constitué exclusivement de modules à base de circuits intégrés (IC) analogiques (à l'exception de la synthèse par wavetable)
- d'un contrôle purement numérique de ces composants

Le contrôle numérique est réalisé par la collaboration :

- d'une carte embarquée de type Raspberry Pi 3 fonctionnant sur Linux Debian
- d'une interface graphique à écran tactile et éventuellement d'afficheurs à LED permettant d'améliorer l'expérience utilisateur
- d'encodeurs à quadrature situés sur la face avant
- d'un ensemble de convertisseurs numérique-analogique reliés à la carte RPi3 par un bus SPI, et permettant de piloter directement les tensions de commandes des IC's analogiques (gain des VCA, fréquences des VCO et VCF, etc).



2. Architecture générale de l'instrument

Le synthétiseur dont nous nous inspirons ici est basé sur le schéma dit de "synthèse soustractive", c'est-à-dire que l'onde de base produite par un oscillateur est filtrée par un filtre (généralement passe-bas) dont la fréquence de coupure varie au cours du temps afin de reproduire l'atténuation des hautes fréquences au cours du temps dans la plupart des instruments acoustiques. Une fois filtrée, l'onde est ensuite modulée en amplitude afin, ici aussi, de reproduire la variation d'amplitude au cours du temps de la plupart des sons produits par des instruments acoustiques (généralement, une phase d'attaque suivie d'une phase de décroissance relativement rapide, puis une phase de tenue pour les instruments à son entretenu, et enfin une phase de relaxation lorsque l'instrumentiste cesse de produire le son).

Le signal path traditionnel d'un tel instrument est illustré Figure 2.1. Une des premières tâches à envisager sera donc l'implémentation analogique de chaque module fondamental. Pour cela, la marque Curtis propose de nombreux circuits intégrés analogiques réalisant les fonctions des différents modules (VCO, VCA, VCF). Un bon résumé des différents circuits existant ou ayant existé sur le marché des instruments analogiques est proposé ici.

Un tel synthétiseur à synthèse soustractive est composé de trois types de modules :

- les oscillateurs (VCO)
- les filtres destinés à enlever des harmoniques (VCF)
- les amplificateurs pour la génération de l'enveloppe globale du signal (VCA)

2.1 VCO

Les VCO sont des oscillateurs générant des formes d'onde plus ou moins complexes, et dont (historiquement, lorsqu'ils sont analogiques) la fréquence est commandée par une tension. Pour ces modules, on envisage deux implantations :

- une implantation purement analogique à base de circuits OTA "LM13700"

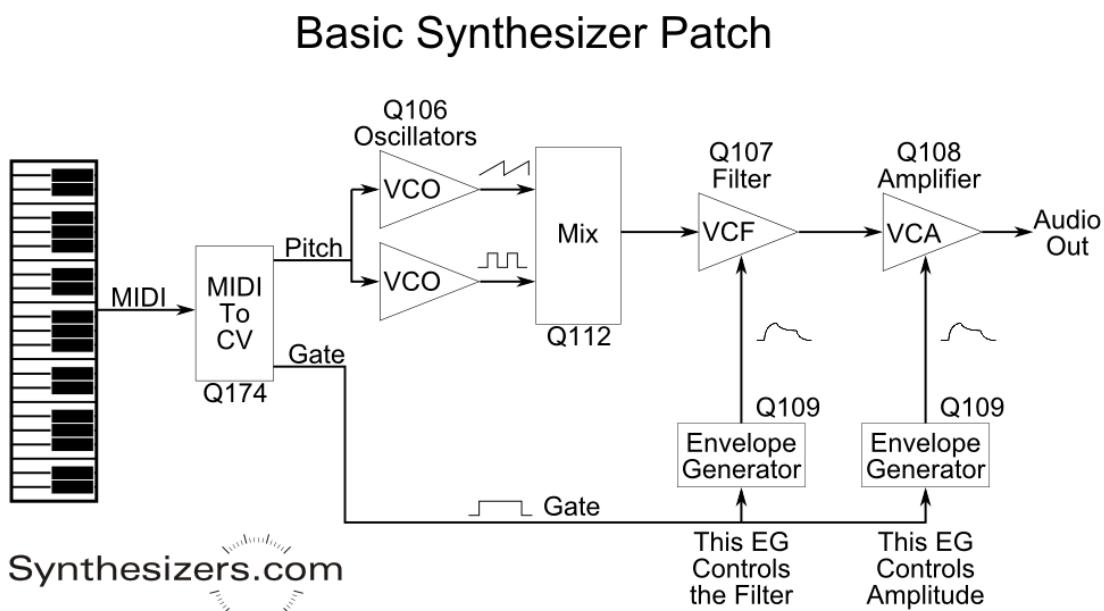


Figure 2.1: Architecture typique d'un synthétiseur analogique à synthèse soustractive : un clavier produit un code MIDI (entre 0 et 127) correspondant à une note de la gamme, cette note est ensuite convertie en une tension (signal "CV") qui est injectée à la fois à l'entrée des VCO (oscillateurs commandés en tension) afin de produire la fréquence (ou "pitch") appropriée et à l'entrée des générateurs d'enveloppe afin de déclencher la production de l'enveloppe de filtre et d'amplitude adéquate.

- une purement numérique consistant à générer les formes d'onde via un convertisseur numérique-analogique et un micro-controleur STM32

L'approche numérique permet de générer des formes d'onde plus complexes, par exemple en utilisant des wavetables, ou par une méthode de synthèse granulaire, tandis que l'approche analogique offre un son généralement plus "chaud" à cause des non-linéarités et des fluctuations du circuit.

2.2 VCF

Les VCF (voltage controlled filters) sont des filtres (généralement passe-bas mais pas seulement) dont la fréquence de coupure est elle aussi pilotée par une tension de commande. On a envisagé deux implémentations :

- à base d'amplificateurs à transconductance (OTA de type LM13700), permettant, via l'architecture SVF State Variable Filter, de produire avec un seul circuit des filtres LP (low pass), BP (band pass), AP (all pass, déphasage) et HP (high pass) ;
- soit à base de circuits intégrés de type CEM3320, qui implémentent déjà un choix particulier d'architecture de filtre.

L'objectif de ces modules VCF est d'ôter des harmoniques au signal au cours de l'évolution temporelle du signal, afin de rendre compte de l'atténuation naturelles (dans les instruments acoustiques) des hautes fréquences. Ainsi par exemple, on utilise un filtre passe-bas pour simuler l'extinction d'une note de piano ou au contraire le régime transitoire initial d'un son de cuivre. De nombreux schémas sont proposés sur le site du fabricant electric-druid.

2.3 VCA

Les VCA sont des amplificateurs commandés en tension (voltage controlled amplifier) dont le rôle est de moduler l'amplitude du signal sonore au cours du temps afin de rendre compte des variations naturelles d'amplitude sonore au cours de la "vie" d'une note produite par un instrument (cf. Figure 2.2). Sur la plupart des instruments acoustiques, la production physique d'une note implique le démarrage (A:attack) puis la stabilisation (D:decay) d'un oscillateur (corde vibrante, membrane, onde sonore dans un tuyau), le maintien plus ou moins long de l'énergie de l'oscillateur si le son est entretenu (S:sustain, cas des instruments à vent ou à corde frottée), puis enfin un régime libre (R:release) au cours duquel l'énergie du système vibrant se dissipe par rayonnement acoustique. Ici aussi on a envisagé des solutions à base de circuits OTA.

2.4 Générateurs d'enveloppe ADSR

Ces différents modules VCO, VCF et VCA sont (pour leur version analogique) pilotés par des générateurs d'enveloppe (attaque, decay, sustain, release). Ceux-ci ont pour rôle de produire les modulations appropriées à la fois des fréquences de coupure des VCF et des enveloppes d'amplitude des VCA ; il arrive très souvent que les enveloppes soient distinctes pour les VCF et les VCA : ainsi pour un son de trompette, l'attaque du VCA est beaucoup plus courte que celle du VCF ; ici, nous envisagerons exclusivement une implémentation numérique.

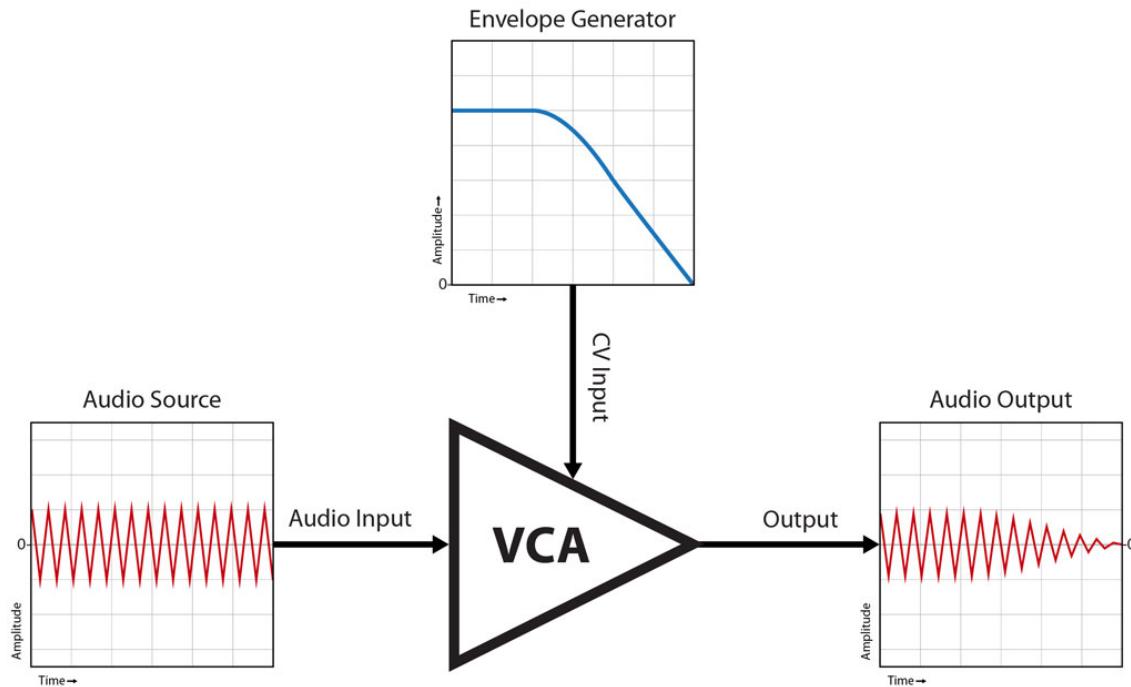


Figure 2.2: Utilisation typique d'un VCA et d'un générateur d'enveloppe pour moduler l'amplitude d'un oscillateur et donner au son produit un caractère plus réaliste : la source audio est le signal brut produit par le VCO (sinus, triangle, carré, etc) ; le signal transite par un amplificateur contrôlé en tension (VCA) par le générateur d'enveloppe via le contrôle CVInput, et produit le signal modulé "Audio Output".

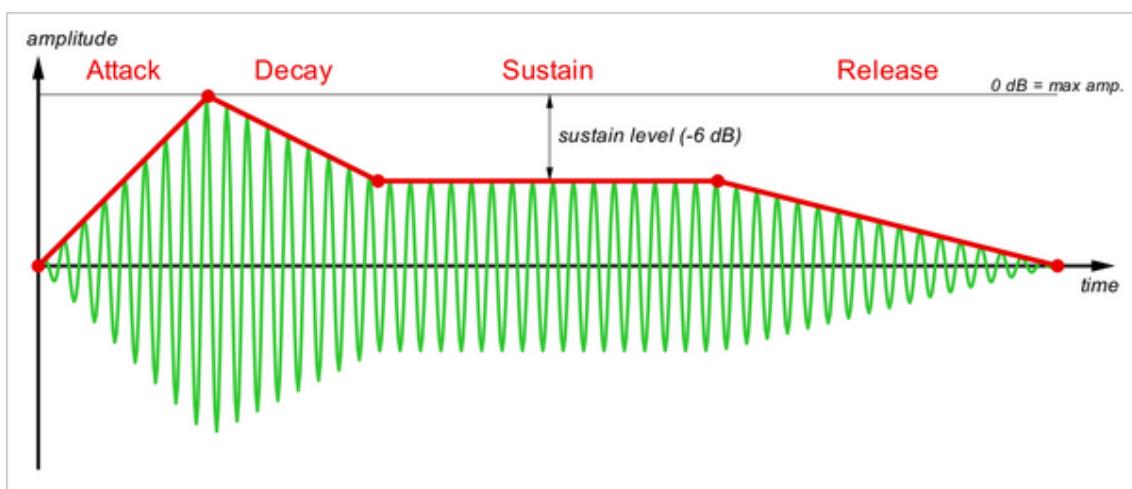


Figure 2.3: Enveloppe typique d'amplitude de type ADSR (en rouge) et signal audio résultant (en vert).

La figure 2.3 illustre une forme typique d'enveloppe de contrôle d'amplitude. Les générateurs d'enveloppe sont chargés de produire une telle enveloppe, paramétrée traditionnellement par quatre paramètres : la durée d'attaque, la durée de décroissance (Decay), le niveau de tenu (Sustain) en dB (ici -6dB à titre d'exemple), et enfin la durée de retour à 0 du signal après relâchement de la touche du clavier (Release time). L'enveloppe ainsi générée vient contrôler le gain d'un VCA (si l'on souhaite contrôler l'amplitude du son) ou d'un VCF (si l'on souhaite contrôler la fréquence de coupure du filtre au cours du temps).

Les enveloppes qui vont piloter les différents modules contrôlés en tension (VCF, VCA, VCO) étant des signaux basse fréquence (en gros, quelques Hz maximum), on utilise un convertisseur numérique-analogique (via bus série SPI) ; nous proposons d'utiliser des CNA de la famille Microchip 4822 connectable en mode SPI.

2.5 LFO et modulations

Pour conclure sur le signal path, ajoutons que des LFO (low frequency oscillators, quelques Hz à quelques dixièmes de Hz) sont indispensables pour donner "vie" au signal. Ils consistent à moduler les différentes caractéristiques des modules VCO, VCF et VCA avec des signaux de modulations variant lentement à l'échelle du signal (en général, les périodes des LFO tournent autour de quelques secondes). Il est notamment possible de réaliser :

- un tremolo en modulant un VCA
- un vibrato en modulant un VCO
- une wah-wah en modulant un VCF

La possibilité de construire une matrice de modulation complexe est une des clés permettant de générer des sons riches. Elle n'a pas encore été implémentée en 217-2018 et reste à faire.

2.6 MIDI

Concernant la commande de l'instrument par un clavier, elle est réalisé via une interface MIDI. Il s'agit d'une norme numérique de communication entre instruments de musique, qui utilise le protocole série asynchrone (dit protocole "1-wire" : un seul fil transporte les données, et l'horloge est reconstruite à l'arrivée) au débit de 31250 bauds. La norme permet de transmettre numéro de note, durée et vitesse, ainsi que des paramètres non-musicaux via les messages MIDI CC (control change) et MIDI PC (program change). On utilisera les broches de l'interface série (Tx/Rx) de la RPi3, et un optocoupleur de type 6N138 pour implémenter cette interface MIDI.

2.7 Interface Homme-Machine

L'instrument est muni d'une interface homme-machine (IHM) efficace et ergonomique. Elle permet de modifier les différentes paramètres sonores (formes de enveloppes ADSR, taux de modulation des LFO et matrices de modulation, fréquences des filtres, effets audio, etc) à l'aide de :

- un écran tactile "touchscreen" piloté par une carte RPi3
- des encodeurs à quadratures

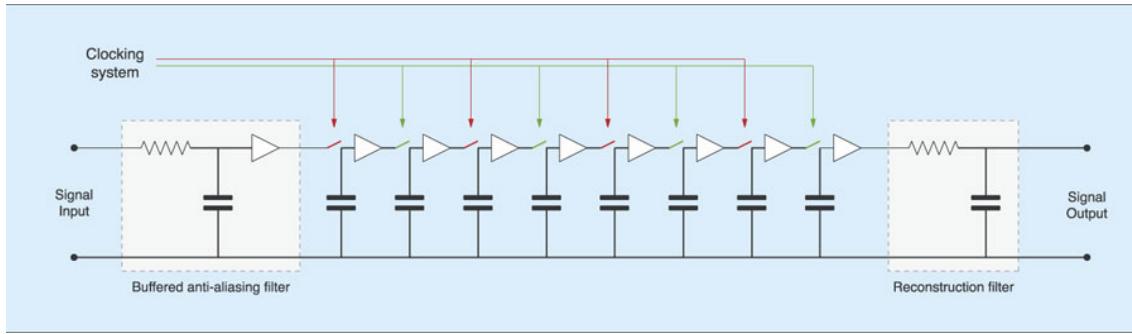


Figure 2.4: Le module de retard est réalisé grâce à une ligne à retard analogique (BBD, Brigade Bucket Device).

- des bargraph à LED se trouvant à côté de chaque encodeur, destinés à montrer la valeur des paramètres ajustés.

2.8 Module d'effet analogique

L'instrument est muni d'un module d'effet de sortie de type delay/chorus à ligne à retard analogique de type BBD. Celle-ci a plusieurs fonctions :

- être utilisé en tant que delay pur
- être utilisé comme chorus/flanger en combinaison avec un modulateur LFO à STM32

2.9 Module de distorsion à tubes

L'instrument est également équipé d'un module "distorsion à tube" capable de colorer le son de sortie du synthétiseur en lui ajoutant des harmoniques caractéristiques des amplificateurs à tubes (=triodes), avec un taux de distorsion ajustable. Ce module est basé sur le composant Nutube (cf. figure 2.5), un nouveau composant fabriqué par Korg et qui intègre deux triodes sur un chip de petite taille, et qui — nouveauté appréciable — autorise des tensions de polarisations "raisonnables", autour de quelques dizaines de Volts (là où des triodes/pentodes classiques exigent plusieurs centaines de Volts) ; la datasheet est disponible ici.

Tous les paramètres (gain, couleur, distorsion, etc) sont programmables par la carte RPi3 ; l'implémentation utilise des potentiomètres numériques programmables de type Microchip MCP4251. Le module est débrayable via l'interface graphique de la RPi3.



Figure 2.5: Le nouveau composant "Nutube" intègre deux triodes de préamplification dans un circuit intégré. Par rapport à des triodes/pentodes classiques, ce circuit permet des tensions de polarisation beaucoup plus faibles (autour de 30 à 80 Volts au lieu des 200 à 400V requis pour une triode classique).

- 11.1 Cahier des charges
- 11.2 Architecture du préamplificateur
- 11.3 Ecart au cahier des charges et du prototype

12 Module delay par A. Mou A. Soler

- 12.1 Fonctionnement Général
- 12.2 Fonctionnement du circuit
- 12.3 Réalisation

A La Java Native Interface

B Code C et Java produites en 2018 (RPi3)

- B.1 Code création des tables d'ondes

C Schémas et datasheets

La synthèse par Wavetable est une technique employant des formes d'ondes périodiques arbitraires afin de générer des sonorités complexes. A la fin des années 70 apparaissent les premiers instruments utilisant cette méthode. Elle se démarque par des sonorités inédites et est utilisée aujourd'hui comme base au fonctionnement de certains simulateurs de synthétiseurs.

Notre approche de la synthèse par Wavetable consiste en l'utilisation des multiplexeurs du synthétiseur afin de choisir entre les son synthétisés par son circuit et les son pouvant être générés via la Wavetable. Il aussi possible de mélanger les deux types de son susnommés pour en créer de nouveaux. Les Wavetables étant éditables, sky is the limit.

3.1 Création des tables d'onde (N. Meerun)

Afin de créer une telle table d'onde, il a fallu tout d'abord se poser des questions, notamment la problématique principale de cette partie:

- **Comment créer des formes d'ondes numériquement?**

Cette problématique principale a été rapidement réglé grâce à une idée plutôt simple:
j'ai eu l'idée d'extraire des échantillons d'ondes à partir de fichiers audio.

Ainsi d'autres questions sont donc apparus, notamment :

- **Quel type de fichier audio utiliser?**
- **Comment créer une forme d'onde à partir d'échantillons audio?**

À la première question, je réponds: **le format WAV**. En effet, ce format présente de nombreux atouts par rapport aux autres formats (MP3 par exemple), nécessaires pour la synthèse audio. On peu citer notamment:

- **Un son beaucoup plus complet, car c'est un format qui n'a subi aucune compression**
- **En raison de sa non-compression, la qualité audio est supérieure**

- c'est un format facile à éditer, notamment en raison de sa structure en bloc de données conformément au *Ressource Interchange File Format: le RIFF*



Figure 3.1: **MP3 contre WAV**. Le format mp3 est un format compressé, ce qui signifie qu'il y a des pertes d'échantillons par rapport au format WAV. En revanche cette perte d'échantillons est largement compensée par sa taille de fichier, qui est nettement inférieure à celle d'un fichier WAV

Le **RIFF** qu'est ce que c'est ? Il s'agit d'un modèle, ou une méthode, qui définit l'organisation interne d'un fichier multimédia, en l'occurrence pour nous, un fichier WAV. Un fichier respectant la norme RIFF a une structure en bloc de données (les "chunks"), **et c'est ce point qui est essentiel pour répondre à notre 2^{eme} question.**

En effet le format WAV respecte la structure du RIFF, et c'est grâce à cette structure, appelée *entête du fichier WAV, ou encore WAVE HEADER* (représenté à la figure 3.2), que nous pourrons extraire les données audio utiles afin de créer notre forme d'onde.

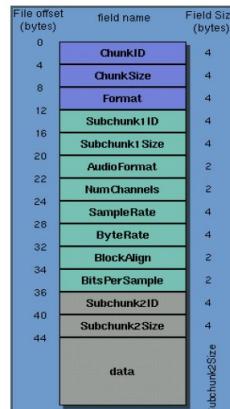


Figure 3.2: L'entête d'un fichier WAV: Chaque bloc de donnée à un rôle: en violet: il s'agit principalement de la description du type de format (en l'occurrence WAV), en vert: ce sont les caractéristiques du fichier audio (nombre de bits/secondes...), et enfin en rose: ce sont les données audio utiles, c'est à dire les données contenant l'audio.

Nous remarquons donc que les données audio sont exprimées en octets, et c'est grâce à ces octets que nous pouvons générer les formes d'ondes grâce à la technique du **rebouclage**, et bien sûr avec un peu de programmation en langage C.

3.2 Le rebouclage en C

Le principe du rebouclage est assez simple: à partir d'un fichier audio (fichier WAV), on isole une partie du son (plus précisément, du signal audio qui est sous forme d'octets), en définissant un **index de début et de fin**. Cette partie étant isolé, on la périodise sur elle-même (d'où le terme de rebouclage), créant ainsi une nouvelle forme d'onde dont la forme peut être plus ou moins complexe. Une illustration de ce principe est donné figure 3.3.



Figure 3.3: On part d'un signal audio (en jaune), et on définit un index de début et de fin (points bleus). La partie du signal audio située entre ces 2 index constituent notre forme d'onde, que l'on reproduira sur elle-même en "boucle"

Après avoir parlé du principe, passons maintenant au codage même de ce principe. Dans cette partie, je ne vais expliquer que les grandes lignes qui permettent d'aboutir à ce résultat (le code C à proprement parlé est présent dans la section "Code C et Java produit en 2017-2018" en annexe).

Tout d'abord, il s'agissait de pouvoir afficher l'entête du fichier WAV passer en argument de ligne de commande (en effet, de un il permet d'avoir les caractéristique du fichier audio, mais aussi et surtout, il permet de s'assurer qu'il s'agit bien d'un fichier WAV): c'est la fonction **void init_buffer_digvco()** qui s'en occupe.

Ainsi, lors de son appel `init_buffer_digvco()`, où `argv[1]` correspond au nom du fichier WAV passer en argument de ligne de commande, le WAV Header (les 44 premiers octets d'après la figure 3.2) va s'afficher, avec l'ensemble des bloc de données ainsi que ses caractéristiques.

L'entête étant affichée, je commence le rebouclage. Nous savons maintenant que les 44 premiers octets sont dédiés à ce dernier, et que les données du son sont situées après. Ainsi je définis donc des index de début et de fin (**loop_start et loop_end dans le code C**), et

je crée une boucle **for** allant de loop_start à loop_end qui stockera les octets dans notre tableau de données (**digvco_buffer()**).

L'indexation créée, il suffit de générer une condition **if** dans une autre boucle, pour que lorsque loop_start dépasse loop_end (à chaque incrémentation de i), loop_start reprenne sa valeur initiale, et par conséquent, digvco_buffer(loop_start) aussi.

J'obtiens donc finalement une forme d'onde finale à partir d'un fichier WAV.

3.3 Génération des ondes numériques "Wavetable", par A. Moutawakil

3.3.1 Choix de la forme d'onde

Ce choix doit se faire sur l'interface graphique du synthétiseur. La version présentée ne permet pas cela et il est nécessaire de modifier le code afin de modifier la forme d'onde choisie. L'information correspondant au choix est envoyée à la carte STM32 via une connexion série (UART) sous forme d'un mot de 12bits, ce mot est donné en argument de la fonction wavetablegen() qui remplit le tableau WAVE LEN avec les échantillons correspondants au signal souhaité. Cet aspect du code n'est pas implémenté dans cette version. Ces échantillons peuvent être générés grâce à une formule mathématique, directement entrés en mémoire ou extraits d'un fichier présent sur un périphérique (deux dernières méthodes en cours de prototypage).

3.3.2 Choix de la note

Quand l'utilisateur appuie sur une touche de son clavier, un signal est envoyé à la carte STM32 une fois encore via son port série (le mot de 12bits précédemment cité). La fonction getnotenumber() intervient à ce moment là en récupérant le numéro de la note envoyée et en la stockant dans une variable n. Cette fonction est aussi conçue de manière à permettre le débogage en cas de problème, une seconde liaison série (USB) permet de consulter les signaux reçus par la carte via un terminal depuis un ordinateur. Il est aussi possible, une fois encore à des fins de débogage, d'envoyer des signaux depuis un terminal, via la liaison USB à la carte.

3.3.3 Écriture de la forme d'onde sur le CNA

Seul un des timers de la carte STM32 est utilisé dans notre approche. Il fonctionne à une fréquence fixe et sert à appeler plusieurs fonctions à intervalle régulier. Afin de modifier la fréquence du signal généré par la carte, il est nécessaire de diviser la fréquence de notre timer. Cette tâche est effectuée grâce à la variable dt qui est calculée à chaque appui d'une touche. dt va servir à incrémenter l'indice de l'échantillon de la Wavetable à envoyer sur le CNA plus ou moins vite, changeant ainsi la fréquence de l'onde générée. En écrivant à chaque appel du timer l'échantillon correspondant à la partie entière de t+dt dans le CNA, il est possible de générer n'importe quelle fréquence inférieure à la fréquence du timer.

L'écriture sur le CNA quand à elle est effectuée grâce à la fonction writeSPI qui envoie sur le bus SPI de la carte STM32 un signal numérique ensuite converti par le CNA.

3.3.4 Code commenté

```
#include "mbed.h"
#include <stdio.h>
```

```

#include <stdlib.h>
#include <math.h>

void set_dt( int note_number);

#define WAVE_LEN 20
#define TIMER_PERIOD 50
#define MS 0.001
#define US 0.001*MS

int flag=0;
float nop_us = 0.5;

double midi_to_freq[127]; // midi_to_freq [note] = frequence de la note

InterruptIn button(USER_BUTTON);

//-----UARTs
Serial pc(USBTX, USBRX); // USB Serial
Serial uart(PA_9, PB_3); // UART Serial

//-----SPI
SPI spi(PA_7, PA_6, PA_5); // mosi , miso , sclk
DigitalOut cs_spi(PB_8);
Ticker timer_spi;

//-----CLKs
DigitalOut CP1(PB_10);
DigitalOut CP2(PA_8);

//-----Variable globales
int counter_wave_table , period_timer_spi;
int note_number;
int wave_table[WAVE_LEN];
float dt;
float t;

void wave_table_gen(char type)
{
    int i;
    if (type == 'r') { // parabolique
        for (i=0; i<WAVE_LEN; i++) {
            wave_table[i] = 10 * i * i;
        }
    } else if (type == 's') { // triangle
        for (i=0; i<WAVE_LEN; i++) {

```

```

        wave_table[ i ] = i * 200;
    }
}
}

void get_note_number()
{
    if( pc.readable() ) { // entree clavier
        // uart.putc(pc.getc());
        set_dt(pc.getc());
    } else if(uart.readable()) { // soit octet dispo sur entree UART RX
        int n = uart.getc();
        set_dt(n);
        pc.putc(n); // debug !
    }
}

void init_freq_midi_tab()
{
    double A=440; // calcul freq base sur celle du La
    for (int i=0; i<128; i++) {
        midi_to_freq[i]=(A/32.0)*(pow(2.0, ((i-9)/12.0)));
        pc.printf(" note %d => freq %f\n", i, midi_to_freq[i]);
    }
}

void writeSPI(int word12bits)
{
    cs_spi=0;
    spi.write(((word12bits >> 8) & 0x0F) | 0x30);
    spi.write((word12bits & 0xFF));
    wait_us(2);
    cs_spi=1;
}

void set_dt(int note_number)
{
    float sigfreq = midi_to_freq[note_number];
    dt = sigfreq * WAVE_LEN * TIMER_PERIOD * US;
    pc.printf(" dt=%f\n", dt );
}

void timer_isr()
{
    // 1) ecriture sur le CNA
}

```

```

t = t + dt;
// t++;
if (t > WAVE_LEN) t -= WAVE_LEN;
writeSPI(wave_table[(int)t]);
// 2) écrire les horloges

if (flag==0) {
    CP1=!CP1;
    wait_us(nop_us);
    CP2=!CP2;
    flag=1;
} else {
    CP2=!CP2;
    wait_us(nop_us);
    CP1=!CP1;
    flag=0;
}
}

int main (void)
{
    CP1=0;
    CP2=1;

    init_freq_midi_tab();
    spi.format(8, 0);
    spi.frequency(1000000);

    set_dt(40);

    wave_table_gen('r');
    timer_spi.attach_us(&timer_isr, TIMER_PERIOD); // us
    while(1) {
        get_note_number();
    }
}

```


4.1 Généralités

Le VCO (*Voltage Controlled Oscillator*), est l'élément de base de la synthèse, qu'il soit analogique ou numérique. Il crée une forme d'onde spécifique dont la fréquence et la hauteur sont commandées par une tension, le *Voltage Control* (Tension de contrôle). Cette tension va provenir, par le biais d'un convertisseur numérique/analogique (CNA), de l'appui d'une touche sur le clavier MIDI du synthétiseur : la hauteur de la note jouée correspond donc au palier de tension correspondant à celle-ci selon une table Tension/Fréquence définie en amont. Le choix d'un contrôle en tension provient du fait qu'un signal électrique servant de commande à l'oscillateur est le moyen le plus souple et rapide.

La sortie du VCO peut-être une onde de plusieurs formes. Le cahier des charges de notre VCO nous invite à avoir deux sorties disponibles :

- **la sortie carrée**, une forme d'onde qui comporte beaucoup d'harmoniques très élevées, en raison de ses fronts de montée très raides. Musicalement, ces caractéristiques font du signal, quand il est reproduit par un haut-parleur, un son agressif et cuivré.
- **la sortie triangulaire**, qui en revanche, n'a pas de fronts de montée raides, mais des points de rebroussement où sa pente s'inverse brusquement. Ici, le son qu'il produit s'apparente plutôt à un bois, plus doux et agréable, moins éclatant que celui du carré.

4.2 Étude théorique de l'amplificateur à transconductance

4.2.1 Principe

On utilise principalement la technologie des *amplificateurs à transconductance* (OTA) : en effet, un OTA est le meilleur choix dans notre cas puisque, contrairement à un amplificateur opérationnel, le gain de l'amplificateur (appelé transconductance g_m) peut être modifié directement par un courant de commande et donc régler les paramètres du montage.

Théoriquement, g_m est proportionnelle au courant de polarisation de l'amplificateur I_{ABC} (*Amplifier Bias Current*) qui doit être fourni à l'amplificateur suivant la formule :

$$I_0 = g_m(V_+ - V_-)$$

$$g_m = \frac{I_{ABC}}{2V_T} (\text{mA/v})$$

où $V_T = 25\text{mV}$

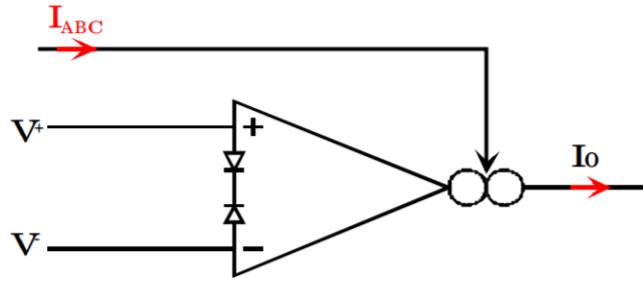


Figure 4.1: Symbole/Modèle d'un OTA

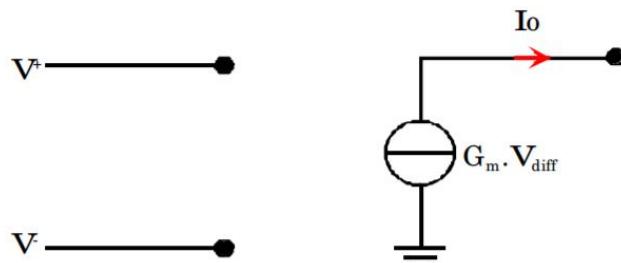


Figure 4.2: Schéma équivalent d'un OTA

L'OTA a une impédance d'entrée Z_i très grande. Le courant de polarisation de l'amplificateur peut être très faible (on peut avoir $I_{ABC} < 1\mu\text{A}$, notamment dans le cas d'un LM13700). On a enfin :

$$BP_{BF} = \frac{\frac{I_{ABC}}{2*V_T}}{2 * \pi * C * A_{BF}}$$

4.2.2 Montages de base

Amplificateur à 1 OTA

Amplification de tension :

$$\frac{V_0}{V_i} = \frac{1 - g_m * R_2}{1 + g_m * R_1}$$

Impédance de sortie :

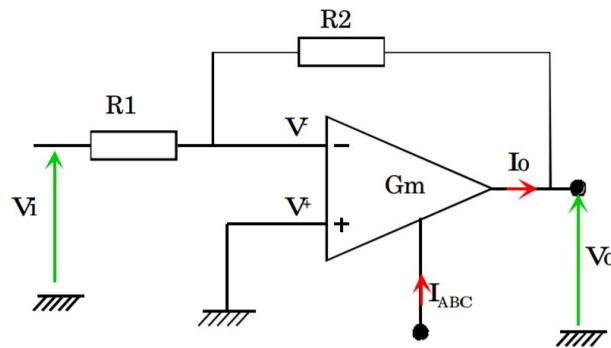


Figure 4.3: Amplificateur de tension inverseur

$$Z_{out} = \frac{R_1 + R_2}{1 + g_m * R_1}$$

Si $g_m * R_1 \gg 1$ alors :

$$\frac{V_0}{V_i} \simeq \frac{R_2}{R_1}$$

$$Z_0 = \frac{R_1 + R_2}{g_m * R_1}$$

Amplificateur à 2 OTAs

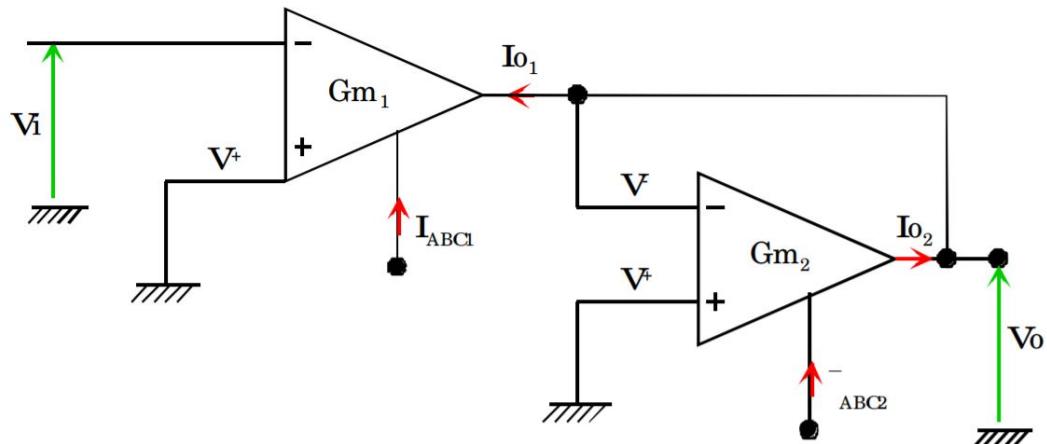


Figure 4.4: Amplificateur à 2 OTAS

$$\frac{V_0}{V_i} = \frac{g_{m1}}{g_{m2}} * Z_{out} = \frac{1}{g_{m2}}$$

On fait varier I_{ABC1} et I_{ABC2} pour l'amplification.

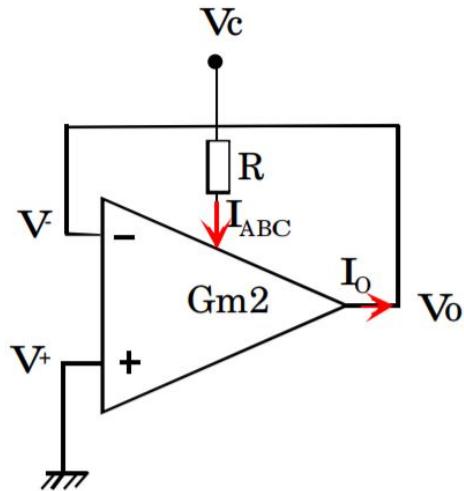


Figure 4.5: OTA à résistance variable

Résistance variable

La résistance est variable en fonction d'une tension ou d'un courant (V_{CR}, I_{CR})

On a :

$$V_0 = V_m$$

$$I_0 = g_m V_m$$

Soit :

$$Z_0 = \frac{I_0}{V_0} = \frac{1}{g_m} = \frac{2V_T}{I_{ABC}} = \frac{2V_T * R}{V_C}$$

Il est possible de concevoir des circuits sans contre réaction (CR). En d'autres termes, au lieu d'utiliser la CR, dont on sait qu'elle réduit la sensibilité de montages aux paramètres (de l'AOP) avec un OTA, il faut simplement considérer la conductance g_m comme un élément de réglage de performances comme le font des composants tels des résistances ou des condensateurs.

Filtre actif du 1er ordre : passe bas à 1 OTA

$$Z(p) = \frac{1}{g_{m2} + C_p}$$

$$\frac{V_0(p)}{V_{in}(p)} = \frac{g_{m1}}{g_{m2}} \frac{1}{1 + \frac{C_p}{g_{m2}}}$$

Filtre actif du second ordre

On peut réaliser des filtres passe-haut, passe-bas, coupe-bandes et passe bandes. La configuration dépend des 3 tensions V_A, V_B, V_C . Ces filtres sont appelés "filtres à fréquence ajustable à Q constant"; en effet, ils préservent la valeur du coefficient de qualité Q sur toute la plage de variation de la fréquence f_0 .

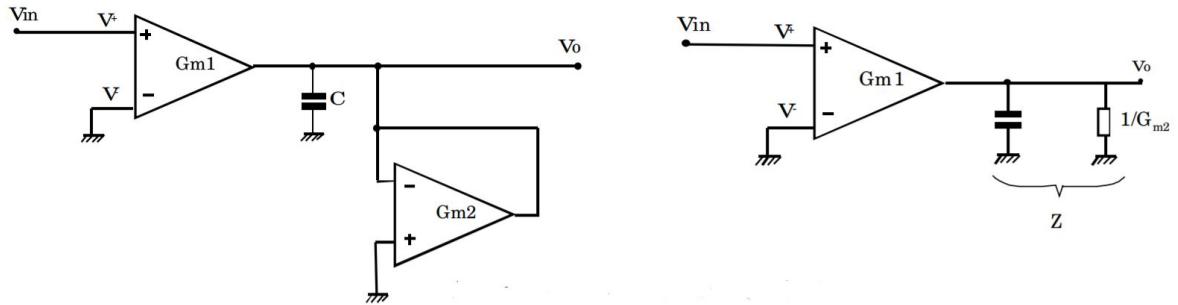


Figure 4.6: Filtre actif du premier ordre

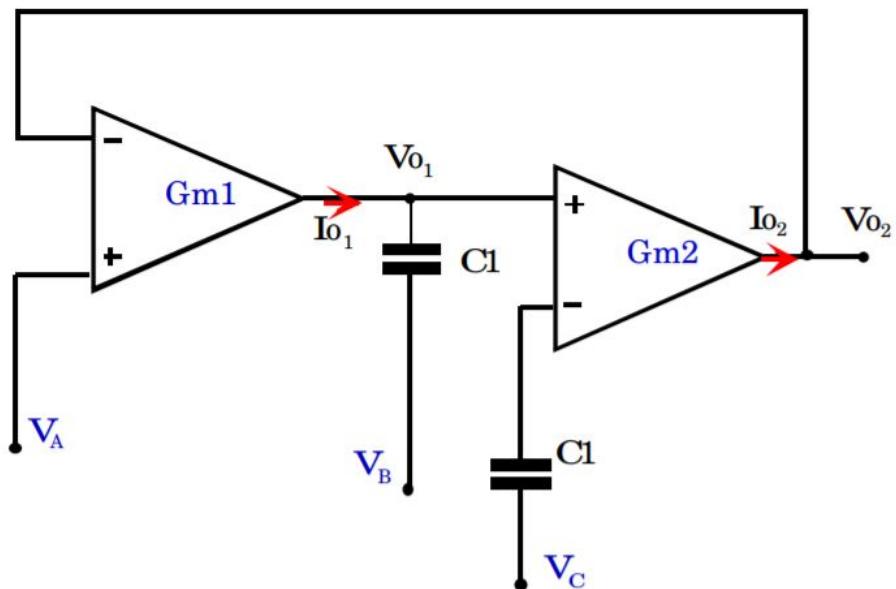


Figure 4.7: Filtre universel du deuxième ordre

$$V_{01} = \frac{g_{m1}g_{m2}V_A + pC_1g_{m2}V_B + p^2C_1C_2V_C}{g_{m1}g_{m2} + pC_1g_{m2} + p^2C_1C_2}$$

Formes particulières :

$V_{IN} = V_A$, $V_B = V_C = 0$ et $g_{m1} = g_{m2} = g_m$

Finalement :

$$\frac{V_{01}}{V_A} = \frac{g_m^2/(C_1C_2)}{p^2 + \frac{pg_m}{C_2} + \frac{g_m^2}{C_1C_2}}$$

4.3 Étude du montage effectué : LM13700

Le principal composant utilisé est donc un **LM13700** contenant 2 OTAs.

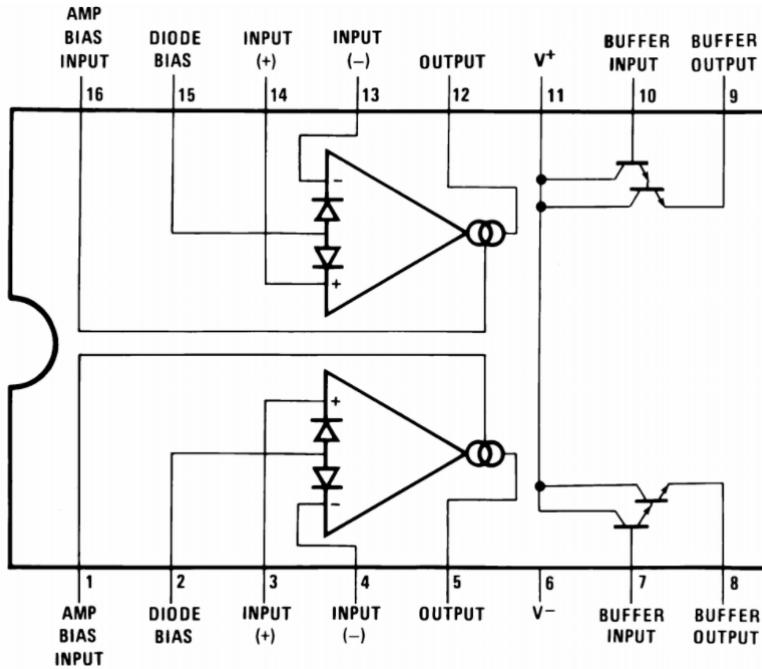


Figure 4.8: Schéma du LM13700

Le montage effectué pour notre VCO est celui proposé sur la figure ??.

La tension V_c permet de régler le courant I_c (la résistance étant définie à $22 \text{ k}\Omega$) et donc, comme on l'a vu avec un montage OTA, la fréquence de l'oscillateur.

On a :

$$f_{osc} = \frac{I_C}{4C I_A R_A} \quad (4.1)$$

Par conséquent, plus la tension V_c est élevée, plus la fréquence de l'oscillateur monte: il faut cependant bien faire attention à limiter cette tension V_c à son seuil maximum déterminé par la limite haute du courant I_c qui est de 1 mA.

La touche jouée par le clavier MIDI est donc convertie en une tension correspondant à la note et qui, en sortie du CNA, passe par un translateur de tension utilisant l'amplificateur opérationnel **TL081** (précision dans le chapitre 6) pour enfin correspondre à la tension V_c voulue du montage.

On retrouve les sorties voulues à deux points différents du montage : la sortie triangle se trouve sur l'entrée (-) du deuxième OTA, au point V_{01} , la sortie triangle est prise au point V_{02} en sortie du buffer de cet OTA.

4.4 Montage avec mélangeur

Pour pouvoir avoir une plus grande variété de sons disponibles, un mélangeur a été ajouté dans le montage afin de combiner différentes formes de signal, numérique ou analogique.

Les sorties des deux VCO (carrés ou triangles) sont acheminées vers celui-ci permettant leur somme en un seul qui peut suivre la suite du *signal-path*. On pourra notamment

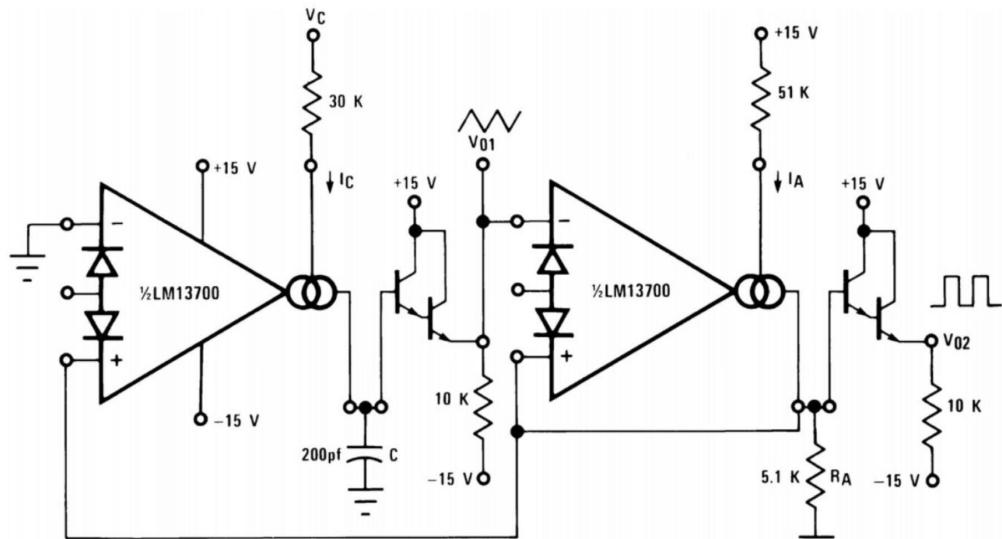


Figure 4.9: Montage d'un VCO à partir du LM13700

régler par la suite les taux de mélange des signaux entrant en appliquant une tension de contrôle sur les différents pins correspondant (voir le chapitre Guide d'utilisation des cartes prototypes) tout en faisant attention à ne pas dépasser les valeurs de l'alimentation du circuit. Ce taux de mélange étant piloté numériquement, il faudra, comme pour les autres modules, un translateur de tension.

Pour créer ce mélangeur, le composant utilisé est un **V2164M** qui contient 4 VCAs (*Voltage Controlled Amplifiers* : voir le chapitre 6) en un seul paquet. On peut donc mettre sur les 4 broches d'entrées les signaux voulus/disponibles (pour l'instant les deux sorties du VCO analogique et la sortie du VCO numérique) pour les sommer en sortie par un amplificateur opérationnel.

La broche 1 du V2164M permet de contrôler le mode de fonctionnement des VCAs : on la relie à la broche 16 (correspondant à l'alimentation +15V) par une résistance de 7,5 kΩ pour avoir un fonctionnement en classe A. (Voir datasheet du V2164M en annexe)

On obtient finalement un montage similaire au schéma proposé figure 4.11

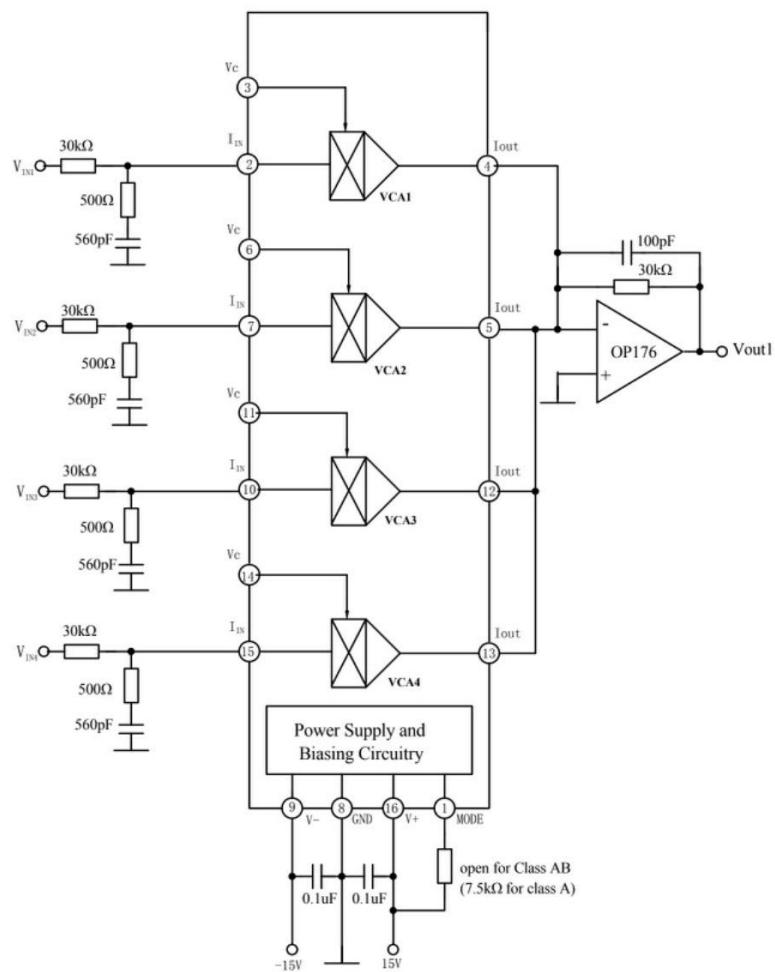


Figure 4.10: Schéma du montage du mélangeur

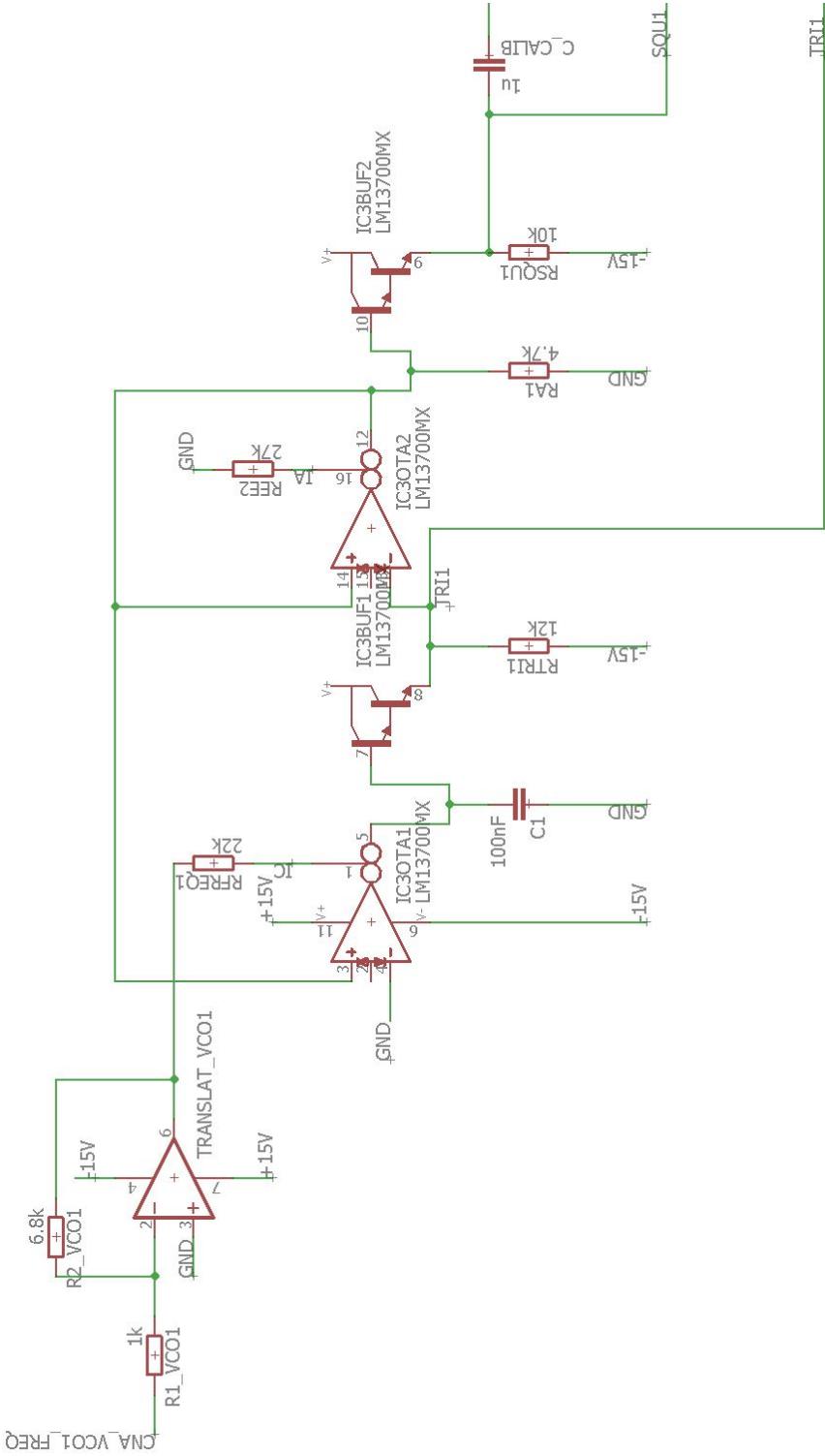


Figure 4.11: Schema Eagle du montage correspondant à la partie VCO

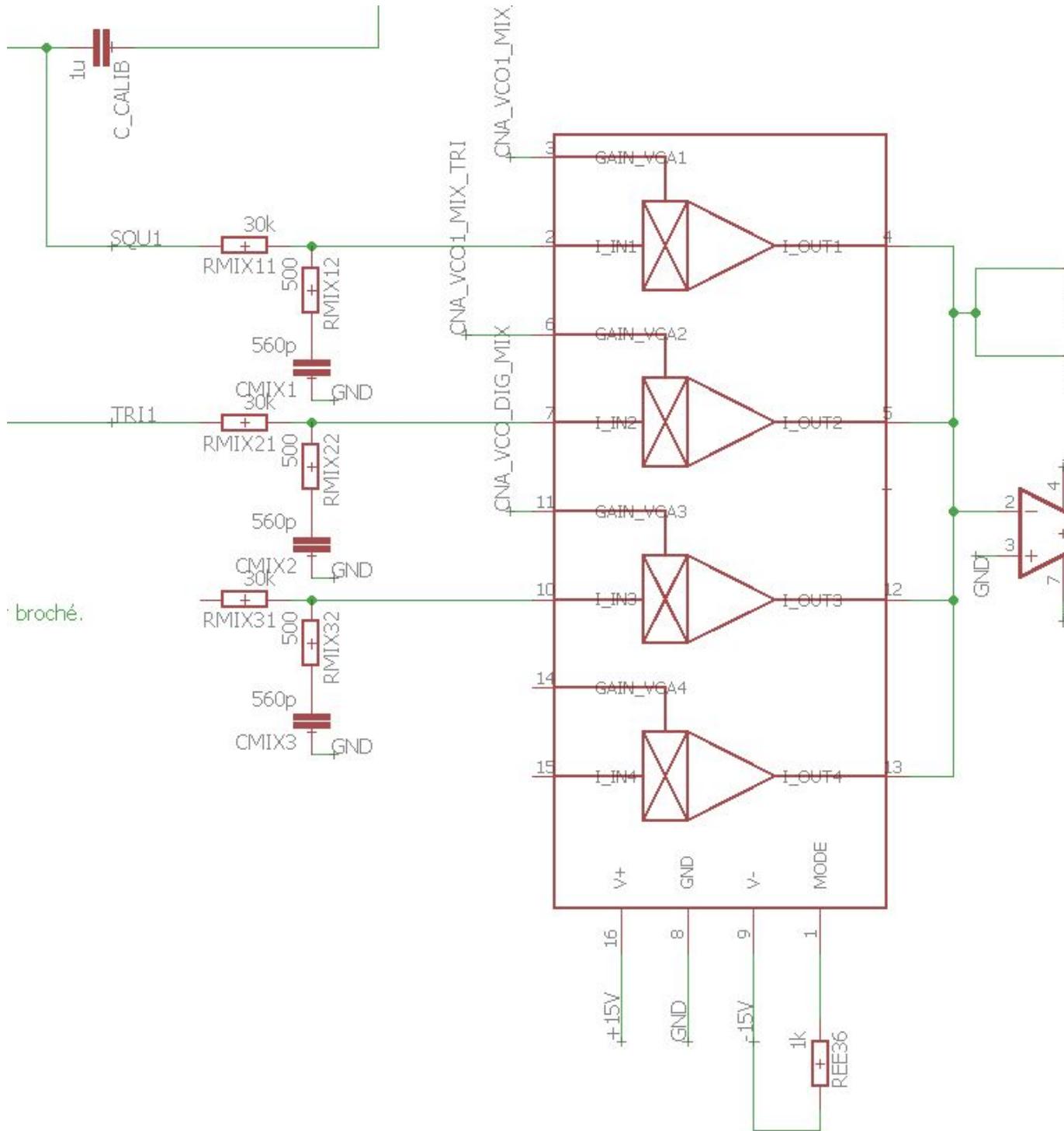


Figure 4.12: Schéma Eagle du montage correspondant à la partie mélangeur

5.1 Généralités

Comme expliqué dans le cahier des charges, PULSAR se base sur une synthèse sonore soustractive : on crée un signal riche en harmoniques qu'on filtre selon notre convenance. Par exemple, si le signal produit par les oscillateurs est un signal carré, on peut le filtrer jusqu'à ne conserver que le fondamental.

Pour cela, nous avons choisi de concevoir des Voltage Controlled Filters (VCF), ce qui permet de commander plusieurs paramètres de nos filtres en changeant simplement les tensions de contrôle :

- la fréquence de coupure
- la fréquence de résonance

Les tensions de contrôles sont générées par des CNA pilotés par la carte Raspberry.

5.2 Caractérisation des filtres

Nous souhaitons pourvoir faire varier notre fréquence de coupure sur une plage de [30Hz; 1500Hz]. Quelque soit le montage, on utilise des filtres RC, on prend des condensateurs avec $C = 300\text{pF}$. Les composants actifs sont alimentés symétriquement en +15V / -15V.

Nous avons donc opté pour un nouveau composant : le CEM3320. Celui-ci était couramment utilisé dans les synthétiseurs des années 80.

Nous configurons nos filtres en passe-bas d'ordre 2 selon le schéma de la figure 5.1.

Avec ce montage, la tension de contrôle de la fréquence influe directement sur celle-ci selon la formule donné par la datasheet du composant:

$$f = \frac{Ae^{-V/Vt}}{2\pi RC} \quad (5.1)$$

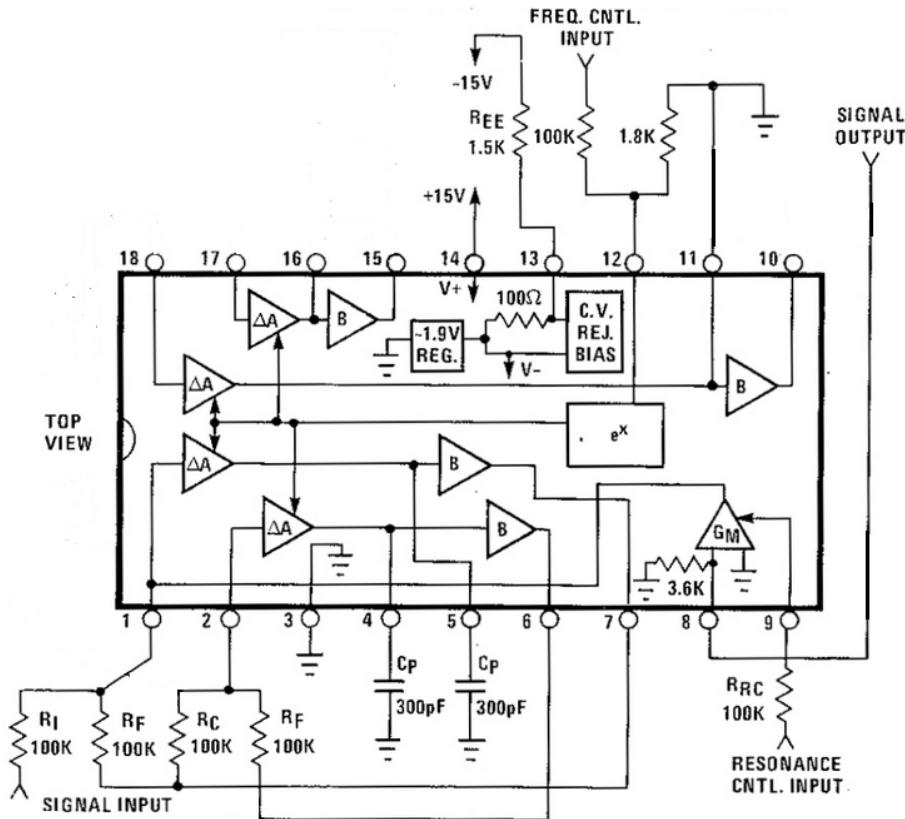


Figure 5.1: Filtre Passe-bas d'ordre 2 à partir d'un CEM3320

V étant la tension de contrôle de la fréquence (broche 12), $A = 0.9$ le gain en courant d'une cellule RC pour $V = 0$, $V_t = 25mV$ et R est la mise en parallèle de $R_f = 100k$ et d'une résistance de $1M\Omega$.

Le principe de ce projet étant de réaliser un synthétiseur hybride analogique-numérique, nous avons fait en sorte que le signal path soit entièrement analogique mais que ses différents paramètres soient commandés numériquement. En ce qui concerne les VCF, nous contrôlons numériquement leurs fréquences de coupure et leurs fréquences de résonance. Cette commande numérique est réalisée avec une carte Raspberry Pi 3. Cette carte pilote des CNA qui renvoient une tension comprise entre 0V et 2.048mV.

Il nous faut alors adapter cette tension à l'échelle de notre composant soit 0V à 13.6V. Nous avons donc mis en place un translateur réalisé avec un AOP :

Ce translateur étant un inverseur, nous l'avons cascadé dans un second ayant ses deux résistances égales à $1k\Omega$ pour compenser l'inversion de la tension. Un montage non inverseur aurait également pu être utilisé, mais aux vues des résistances disponibles, cette méthode était plus pratique pour contrôler avec précision notre tension de contrôle.

5.3 Conclusions sur l'implémentation des VCF

Nous avons obtenu de bons résultats quant au fonctionnement des VCF : la fréquence de coupure balayait bien la plage de fréquence [30Hz; 1500Hz] et était précisément contrôlable.

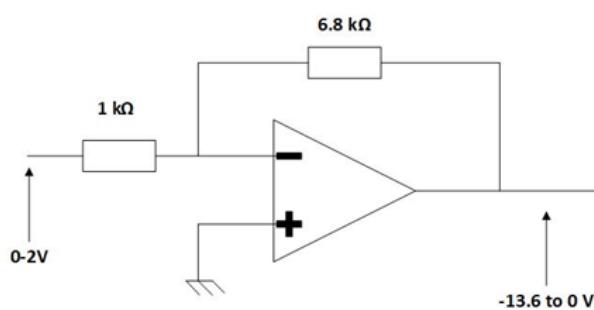


Figure 5.2: Schema du translateur

Son principe de fonctionnement est le suivant : il va imposer une enveloppe au signal sonore qu'il reçoit en entrée grâce à une tension de contrôle. Dans la figure 6.1 on observe un signal triangulaire, qu'on multiplie par une enveloppe lentement variable qu'on appelle "enveloppe timbrale". Ce qui a comme résultat : un signal de sortie va être atténué progressivement jusqu'à son extinction complète.

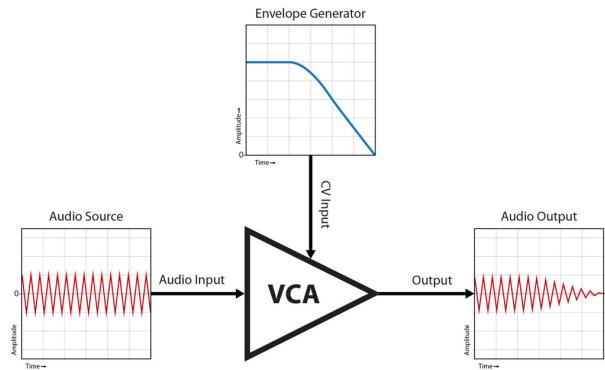


Figure 6.1: diagramme de fonctionnement

Le modèle que nous utilisons est celui de la figure 6.2.

J'ai choisi d'utilisé un OTA (Ampli à Transconductance) LM13700 (voir datasheet en annexe) qui se comporte comme un AOP basique mais avec une tension permettant de contrôler le gain. On cherche à générer une enveloppe de type ADSR, composée de quatre paramètres :

- attack (ms) : temps de montée après appui sur la touche du clavier
- decay (ms) : temps de décroissance jusqu'au plateau après la fin de la phase d'attaque
- sustain (niveau) : niveau du plateau en pourcentage

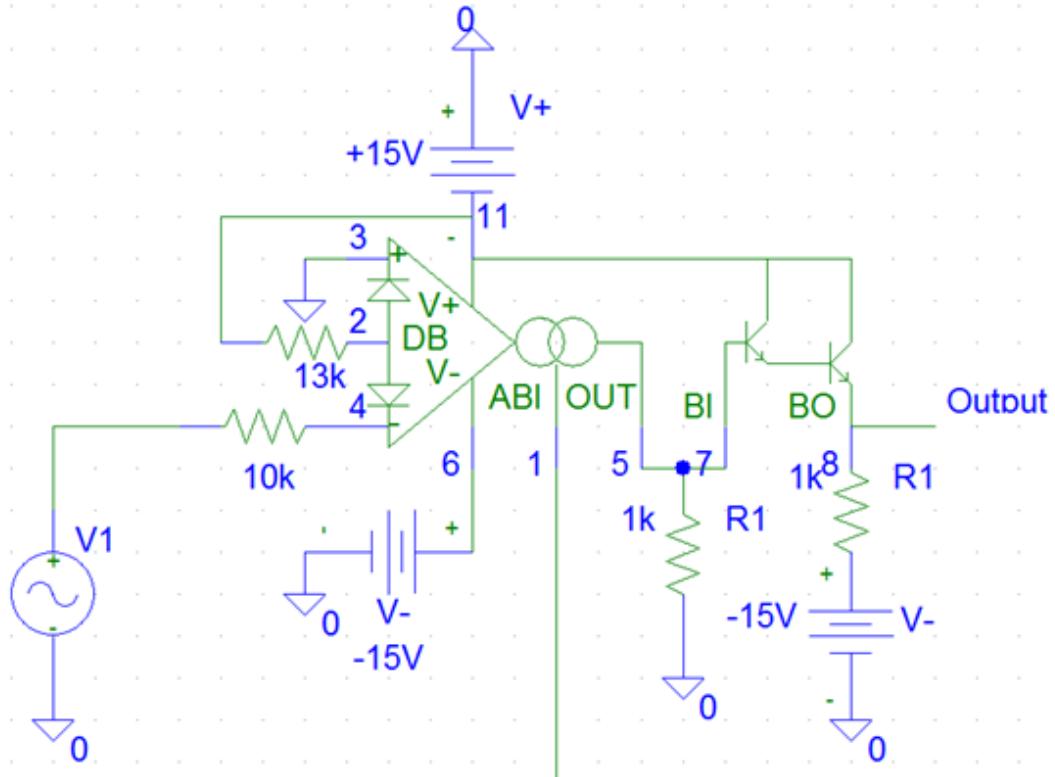


Figure 6.2: Schéma du VCA à LM13700.

- release (ms) : temps d'extinction après relâche de la touche du clavier

On sait que lorsque la tension de contrôle est nulle (sur la patte 1) le signal n'est pas amplifié, c'est à dire que le gain vaut 0dB. et lorsque qu'elle est inférieure à -13.6 V, le signal (et donc le son en sortie jack) sera complètement atténué. Le cahier des charges et l'intérêt du synthé étant l'hybridation, le pilotage de ce composant doit se faire numériquement par une Raspberry. D'où le choix d'un OTA qui permet ce contrôle en tension, puisqu'un OTA est avant tout un moltiplicateur de deux signaux, donc permet de moduler le signal audio avec une tension de contrôle. La carte de contrôle dispose de plusieurs CNA qui délivrent une tension de sortie entre 0 et 2,048 V (2048mV ou 2^{11}).

On constate qu'il y a une différence entre la tension de sortie du CNA et la tension nécessaire à l'atténuation totale du signal. Pour combler cet écart j'ai élaboré un simple translateur qui va permettre de « translater la tension ». On va donc avoir ce translateur qui va nous changer le 0V en 0V et le 2.048V en -13.6V. La sortie de ce translateur arrive sur la patte une du schéma précédent. L'ALI utilisé est un TL081 (ici on n'a pas de contraintes de bruit).

A ce schéma, j'ai ajouté 3 diodes qui permettent de combler un léger gap. En effet j'ai remarqué qu'à cause de l'architecture interne du LM 13700, on a du mal à atteindre les -13.6 V nécessaire et que l'on continuait à entendre le son de la dernière note joué. En ajoutant 3 diodes de 0.6V on diminue encore de 1.8V la tension qui arrive sur la patte 1. On entend encore un léger bruit si le potard de volume de l'enceinte est trop élevé. Ce léger problème va être à régler, soit par l'ajout de diode (Zener peut être) qui risque de prendre plus de place sur le PCB et pas très technique comme solution. L'autre solution

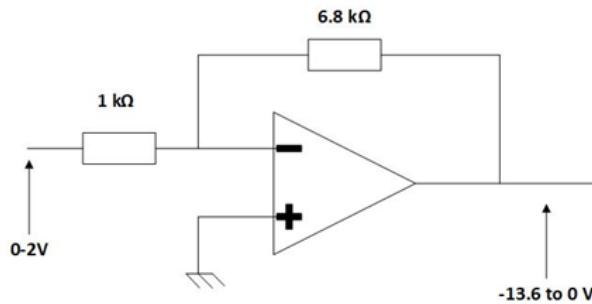


Figure 6.3: Schéma du translateur

peut être de diminuer la résistance R2 du translateur (celle de 6.8KOhms).

Le VCA possède maintenant un pont diviseur de tension muni d'un potentiomètre afin de diminuer l'amplitude du signal entrant ainsi qu'une capacité de découplage qui enlève la composante continue du signal. La sortie de ce pont diviseur de tension a permis notamment de souder un pin qui sert au programme de calibration de la Raspberry. (cf contre rendu de Mounir). En effet le signal d'entrée de ce pont diviseur découpé provient du signal de sortie carré du VCO. En diminuant son amplitude et en le centrant autour de 0 la Raspberry est capable de mesurer sa fréquence, ce qui permettra la calibration du VCO. On peut parler d'un accordeur de synthé.

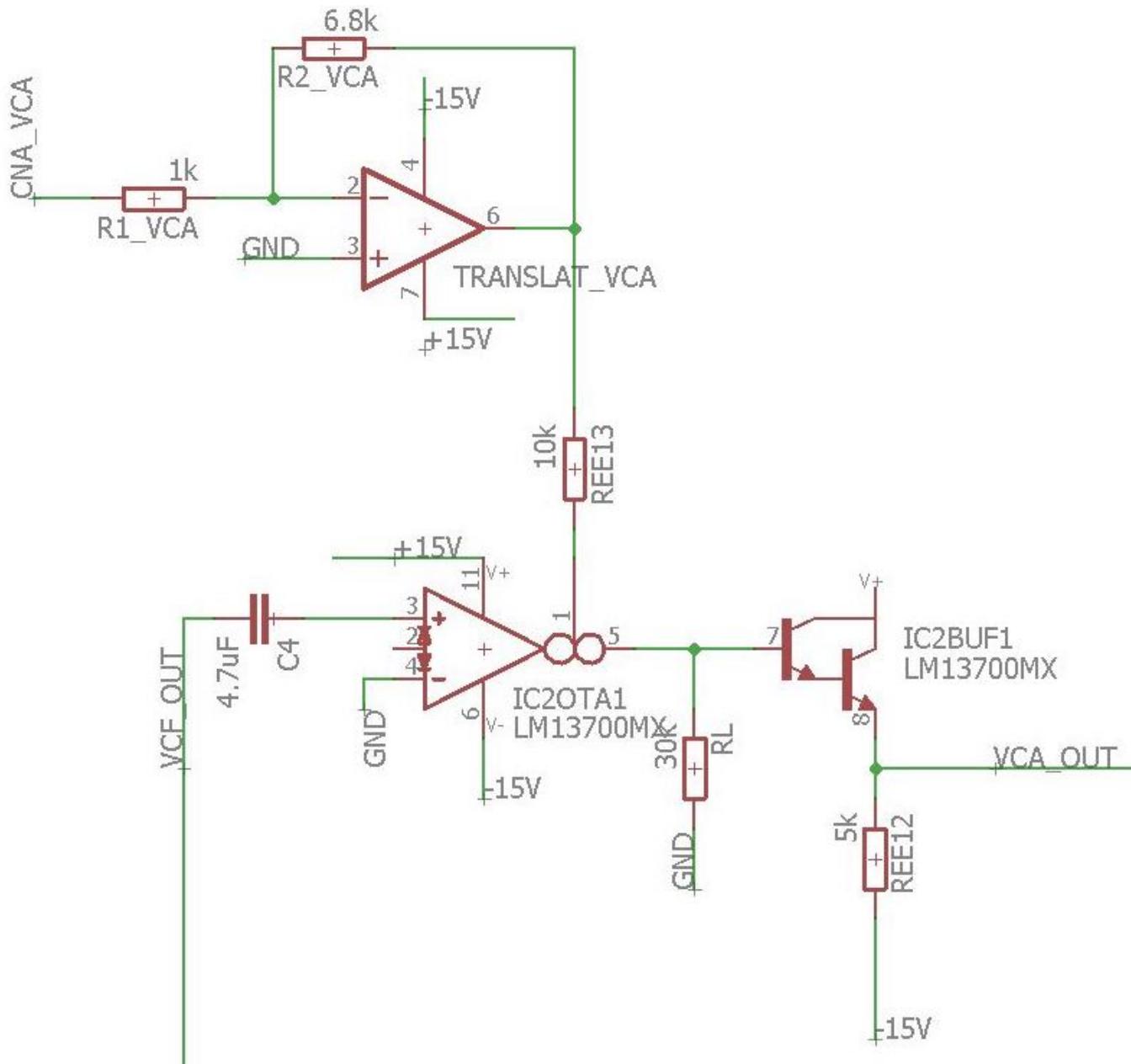


Figure 6.4: Schéma Eagle du montage correspondant à la partie VCA

7.1 Carte analogique (V. Clabaux)

La carte analogique du Pulsar est doté d'une petite carte additionnelle où se trouve un connecteur 10 broches. Ce connecteur permet de diminuer la quantité de fil flottant de la carte. Il assure la communication entre la Raspberry qui quant à elle pilote :

- les taux de mélange des VCO
- la fréquence des VCO
- la fréquence de résonance et la fréquence de coupure du VCF
- la tension de contrôle du VCA

(deux pins sont réservés à la masse afin que les cartes aient une masse commune sans branchement supplémentaires).

Les pins sont disposés comme suit : (à la légende viendra s'ajouter un code couleur concocté par mes soins)

- 1 : Mix VCO 1 sortie triangle (non soudé)
- 2 : CV fréquence VCO 1 (câble violet)
- 3 : Mix VCO 1 sortie carré (non soudé)
- 4 : Mix VCO Numérique provenant de la STM32 (non soudé)

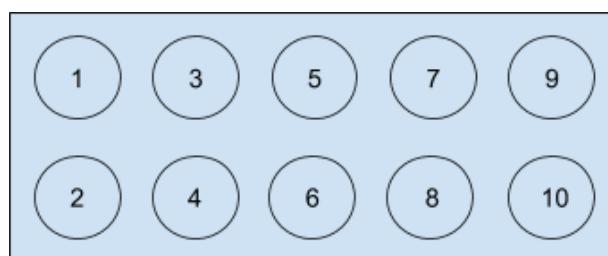


Figure 7.1: pins de la carte

- 5 : CV fréquence de coupure VCF (câble jaune)
- 6 : Sortie carré du VCO 1 (après condensateur de liaison), pour le calibrage
- 7 : CV VCA (câble orange)
- 8 : N/C
- 9 et 10 : GND (fil noir)

La carte est également équipée d'un domino 4 voies qui permet une alimentation moins prise de tête avec, de la ligne 18 vers la ligne 12 : 0 / +5V / +15V / -15V. Ce domino est à utiliser pour l'alimentation de toute la carte, les lignes d'alimentation de la carte sont déjà tirées :

- +15V : ligne 2 et 31
- -15V : ligne 3 et 32
- GND : ligne 4 et 33

Les lignes d'alimentation de la carte secondaire (l'extension dont les lignes ne sont pas numérotées sont dessiné au feutre.

La sortie audio du VCA (qui est celle du module analogique par la même occasion) se fait sur la patte 8 de l'OTA. Un pin femelle y est soudé afin de pouvoir plugger le fil de la prise jack (fil bleu, le fil jaune doit être à la masse) et un pin mâle permet de visualiser la sortie du signal sur l'oscilloscope. Conseil pratique : pensez à mettre l'oscillo en temporel pour visualiser les différentes étapes de l'enveloppe ADSR.

Le 5V n'est pas tiré sur la carte de la team analo puisqu'elle sert à alimenter la Raspberry.

ATTENTION : Penser à vérifier où sont coupées les lignes ! Les coupures sont signalées par des traits verticaux au feutre rouge sur la plaquette. (Voir photos, qui seront bientôt là).

Précautions d'utilisation

L'utilisateur et/ou futur concepteur devra, pour chaque soudure de nouveaux composants, vérifier si la ligne n'est pas utilisée par ailleurs. Si oui, alors il devra sélectionner une autre ligne et si toutes les lignes voisines sont utilisées, il devra sectionner la ligne et l'indiquer au feutre sur le dessus de la plaque, afin que les différents modules ne communiquent pas ensemble. Il devra également penser à séparer les nouveaux modules des modules existants. (Exemple : photo de séparation des différents modules VCO/VCF/VCA.)

Chaque point clé d'un nouveau module devra être accompagné d'un pin test qui permettra de voir rapidement les éventuels problèmes à l'oscilloscope.

Si le VCF ne fonctionne pas correctement, tester chaque étage indépendamment. Si l'un d'entre eux dysfonctionne, penser à adapter le schéma pour utiliser un autre étage du CEM3320.

Important : les OTA sont des composants très peu puissant. C'est pourquoi la limitation de courant doit être réglé à 0.1 A et il faut s'assurer que leur consommation n'est pas supérieure à 0,02A. Pour vérifier si le LM13700 n'est pas cramé il faut prendre la tension sur la patte 1. Et vérifier que l'on a une tension de -13.6V lorsque le composant est alimenté (en -15V). Il y a en effet toujours 1.4V d'écart (= 2 diodes) entre le potentiel de la patte 1 et le potentiel d'alimentation négatif.

7.2 Carte convertisseurs (S. Reynal)

Il s'agit d'une carte contenant essentiellement les CNA chargés de générer les tensions de contrôles pour les circuits analogiques : tension de contrôle des VCO et enveloppes pour le VCF et le VCA. Le design en est documenté dans le schéma EAGLE "cna.sch". Les CNA sont des circuits MCP4822 : double convertisseur sur 12 bits, interface sur bus série synchrone SPI. La carte peut comporter jusqu'à 8 circuits, soit 16 convertisseurs en tout.

L'adressage des convertisseurs (qui sont tous connectés au même bus SPI de la Raspberry) est réalisé par l'intermédiaire d'un décodeur 3-vers-8 74HC138 dont les entrées d'adresse sont connectées à des ports GPIO de la carte Raspberry, et dont les sorties Y_i sont connectées aux entrées "chip select" CS des MCP4822. Il est donc possible d'adresser jusqu'à 8 circuits. Néanmoins, même si nous n'utilisons plus les signaux CE0 et CE1 (ou CS0 et CS1 selon les nomenclatures) du bus SPI de la Raspberry pour adresser les convertisseurs¹, nous utilisons le signal CE0 indirectement parce qu'il a un chronogramme conforme à la norme SPI : CE0 est donc connecté à l'entrée 9 du décodeur (G2B), qui valide la sortie Y_i correspondant à l'adresse présentée sur les entrées A B et C. Ainsi, lorsqu'on écrit sur le device SPI, le chronogramme de la sortie Y_i approprié suit exactement celui du signal CE0.

Le protocole d'écriture d'un mot à convertir est le suivant :

- Positionner les ports GPIO (pins 31 33 et 35 de la Raspberry) connectés aux entrées A B et C du décodeur d'adresse de façon à activer le circuits MCP4822 souhaité
- Ecrire le mot de 16 bits appropriés sur le device SPI "/dev/spi0.0" (soit directement avec un ioctl, soit indirectement avec les bibliothèque Piggpio ou WiringPi)
- Si l'on écrit sur plusieurs CNA et qu'on souhaite synchroniser les mises à jour des sorties de tous les CNA, le signal LDAC est disponible (pin 29) : dans ce cas, il faut alors mettre LDAC à 1 avant d'écrire dans le device SPI, puis descendre LDAC à 0 à la fin du processus d'écriture sur TOUS les CNA.

¹Le bus SPI de la Raspberry ne peut adresser "automatiquement" que 2 circuits, via les signaux "chip enable" CE0 et CE1 du bus SPI0.

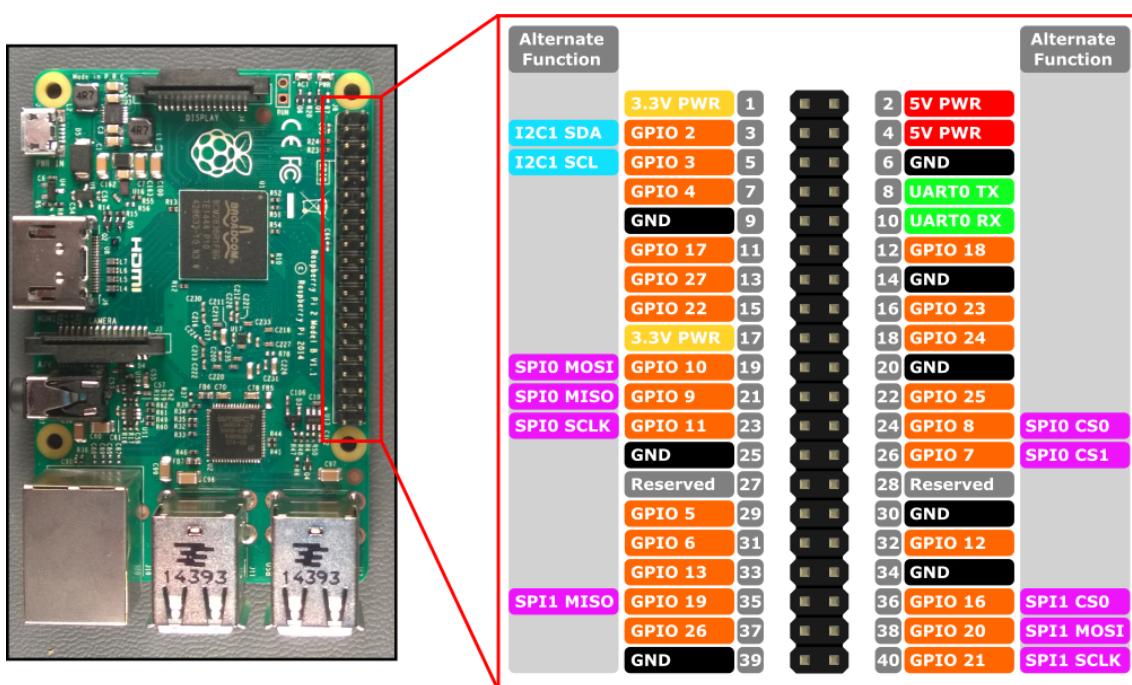


Figure 7.2: Rappel des pins de la carte Raspberry

8.1 Principes

Le principe du synthétiseur s'appuie sur une hybridation analogique-numérique. Il faut donc convertir les signaux numériques envoyés par la Raspberry Pi 3 en signaux analogiques qui contrôlent les VCO, VCF et VCA. Pour ceci, on utilise des microchips MCP4822 (dont la datasheet est disponible sur ce site) qui feront office de CNA (Convertisseurs Numérique-Analogiques) 12 bits car les signaux numériques qui sont envoyés par la Raspberry Pi 3 via le bus SPI (Serial Peripheral Interface) sont des bus de données de 12 bits. Le CNA convertit le signal reçu en une tension électrique de 10mV à 5,5V environ.

La microchip MCP4822 est conçue pour que chacune d'entre elles jouent le rôle de deux CNA (en effet, on a deux sorties CNA-A et CNA-B sur les ports de la microchip représentés en FIGURE 7.1).

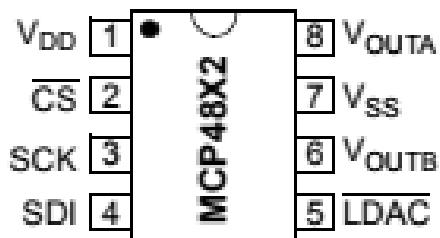


Figure 8.1: Microchip MCP4822 avec ses entrées/sorties

Par ailleurs, les CNA sont pilotés par la carte Raspberry Pi 3 qui envoie les données par la liaison SPI et qui décide aussi de la Clock (entrée SCK du CNA). Les différents ports de la RPi sont représentés en FIGURE 7.2, on voit qu'il y a deux liaisons SPI sur la Rpi, on

peut donc contrôler deux microchips à la fois et donc 4 CNA !

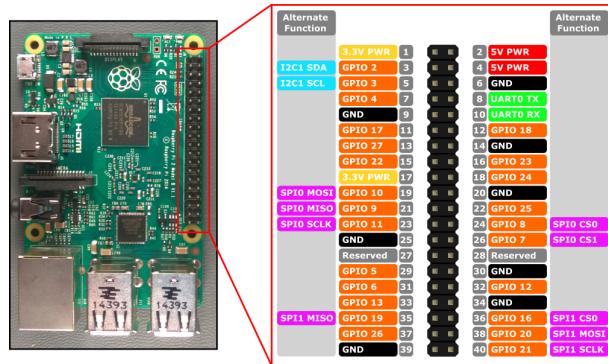


Figure 8.2: Les entrées/sorties de la Raspberry

Le branchement entre la carte et la microchip est donné par :

- VDD 3.3V sur la pin 1 du header
- CS sur la pin 24 du header (CE0 du bus spi)
- SCK sur la pin 23 (CLK du bus spi)
- SDI sur la pin 19 (MOSI du bus spi)
- LDAC sur la pin 12 (GPIO 1)
- VOUT B, la sortie analogique du CNA B
- GND sur la pin 39 (0V)
- VOUT A, la sortie analogique du CNA A

Le rôle de ces CNA est d'envoyer les fréquences de contrôle des VCO mais aussi d'envoyer les amplitudes des enveloppes ADSR (Attack, Decay, Sustain, Release) pour le VCA et le VCF. Ces enveloppes ADSR (dont un exemple est donné en FIGURE 7.3) sont paramétrées par 4 paramètres : la durée d'attaque (Attack) de la note, la durée de décroissance (Decay), le niveau de tenu (Sustain) et la durée de retour à zéro du signal (Release) après relâchement de la touche du clavier.

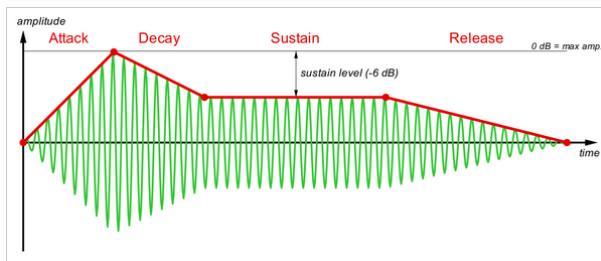


Figure 8.3: Forme typique d'enveloppe d'amplitude

8.2 Génération des signaux numériques par la carte

Pour générer les signaux numériques contrôlant les CNA, on exécute sur la carte un code écrit en C qui permet d'envoyer les bus de données de 12 bits. Afin d'être "compris" par la Raspberry, le code en C doit être écrit à l'aide de bibliothèques qui font le lien avec la carte. La bibliothèque utilisée dans le code est la bibliothèque PIGPIO qui permet d'initialiser les ports GPIO de la carte et de les régler en mode SPI par exemple.

Code pigpio à venir !

9.1 Principes

Dans le cadre de la réalisation de l'interface permettant la communication homme-machine (IHM). On paramètre une commande par encodeurs rotatifs et LEDs bars. le contrôle numérique est ainsi géré par une carte embarqué de type Raspberry Pi3.

Qu'est ce qu'un encodeur à quadrature ?

C'est un composant permettant la transmission d'un certain nombre de pas d'incrément ou de décréments (selon le sens de rotation mécanique choisis par l'utilisateur) vers une carte RPI 3B. Les modèles présents dans le marché varient mais ils délivrent tous deux signaux d'impulsions carrés déphasés de 90 degrés.

Les deux états majeurs que peuvent avoir les signaux en sortie de l'encodeur sont : Signal A en avance par rapport au signal B ou bien signal B en avance par rapport au signal A. Ces deux états décident de la direction de rotation de l'encodeur que l'on va transmettre.

Le tableau introduit les 4 états possibles selon l'état de chacun des signaux qui seront appliqués aux entrées I/O de notre carte embarquée:

Le principe de fonctionnement interne des encodeurs est une suite de fermeture et ouverture de contact mécanique. Le signal issu des deux voies de l'encodeur présente un problème mécanique lors de son acquisition: l'effet rebond dû aux vibrations de la lame de l'interrupteur.

Cet effet peut être supprimé à l'aide d'un glitch filter électrique ou numérique, on choisit d'utiliser la fonction anti-rebonds de la bibliothèque Piggio.

Afin de fournir un code permettant de traduire les mouvements de rotation de l'encodeur en signaux qui seront traités par les différentes parties du synthétiseur, on crée l'algorithme suivant ces étapes :

- Filtrer le signal d'entrée par un glitch filter afin de supprimer les rebonds.
- Définition d'une machine à état traduisant les changements d'état des signaux d'entrées.

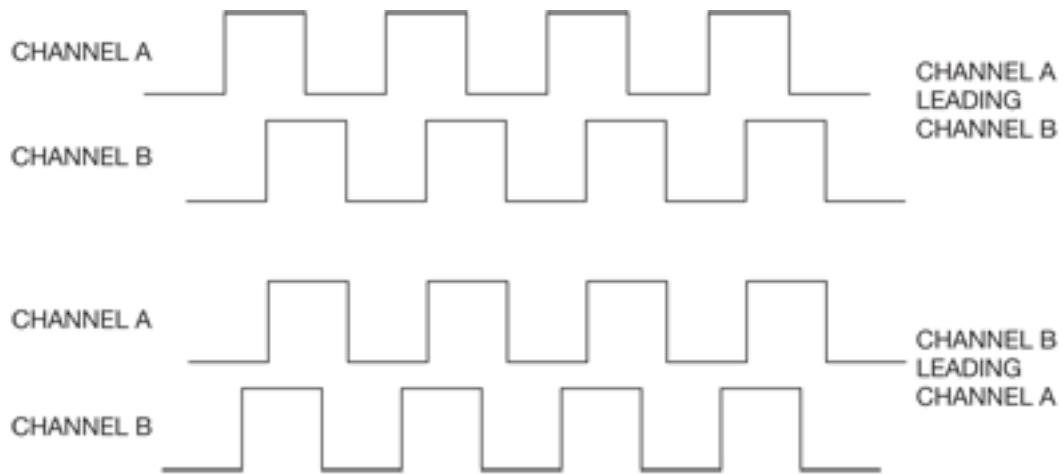


Figure 1 A quadrature encoder produces pulses that are 90° out of phase.

Figure 9.1: Exemple de signal d'impulsions issus d'un encodeur à quadrature

TABLE 1 LOGIC STATES OF QUADRATURE ENCODER

	Channel A	Channel B
State 0	0	0
State 1	1	0
State 2	1	1
State 3	0	1

Figure 9.2: Table des différents état logiques d'un encodeur à quadrature

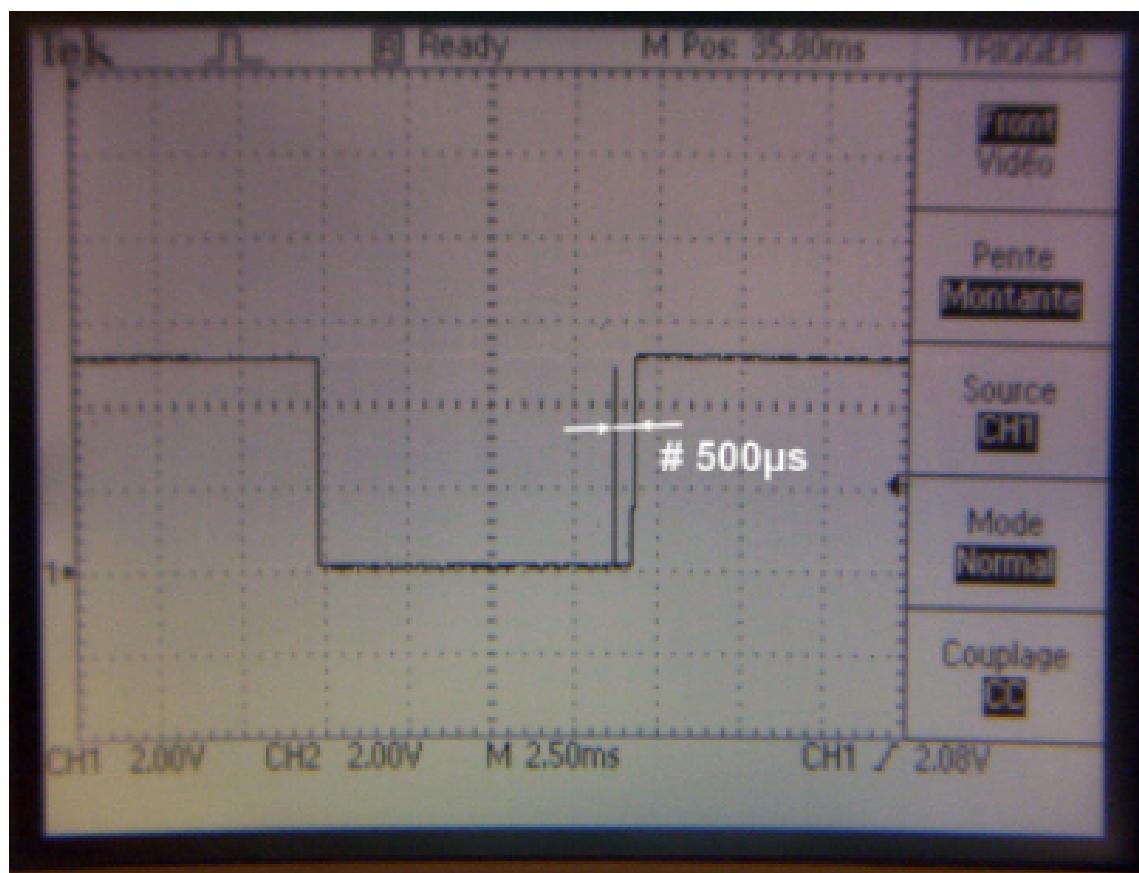


Figure 9.3: Exemple de rebond aléatoire sur un front montant d'une voie.

- Création d'une fonction de comptage qui sera incrémenté ou décrémenté selon le sens de rotation.
- Mesure de l'état présent des deux voies et leurs comparaison avec l'état précédent.

Afin d'optimiser la gestion des différentes informations pour chaque encodeur on crée une structure réunissant le numéro de broche de l'Expander lié à chaque voie de l'encodeur ainsi que leur niveau logique, la position courante et précédente de l'encodeur ainsi qu'une variable enregistrant la dernière voie à avoir changé de valeur.

On commence par initialiser l'encodeur, une première partie du code nous permet d'allouer une mémoire suffisante pour mémoriser les différentes informations qu'on récupère via notre structure, puis de définir les différents numéro de broche pour chaque voie de chaque encodeur.

L'étape suivante est de détecter l'encodeur ayant changé de position, puis détecter la voie sur laquelle il change de position et comparer ces informations avec ceux enregistrées précédemment pour pouvoir reconnaître le sens de rotation. après cela on met à jour les variables de positions dans la structure afin d'enregistrer le dernier mouvement de l'encodeur puis en comparant les données collectées on incrémente ou on décrémente la valeur de l'encodeur, le résultat final de ce code est récupéré afin d'allumer les LEDs Bars en fonction de la position de l'encodeur.

9.2 Pourquoi des Leds Bars ?

Une suite de dix Leds rouges positionnées autour de chaque encodeur afin de permettre une meilleure visualisation de sa position actuel. Reliées à la carte embarquée, ils reçoivent selon l'état de l'encodeur un signal de sortie à l'état haut ou bas. Ces leds nécessitent un plus grand courant que celui fourni par les sorties GPIO de la carte, pour cela on utilise deux circuits ULN2803A contenant chacun 8 darlintons et permettant de résoudre l'inconvénient du faible courant de sortie des GPIOs.

9.3 Qu'est ce qu'un expander MCP23017?

L'expander MCP 2307 (cf. datasheet en annexe) consiste en des registres de configuration multiples 8 bits pour la sélection d'entrée, de sortie et de polarité. Le système maître de cet expander est notre Raspberry PI 3, ce dernier permet d'activer les E/S comme entrées ou sorties en écrivant sur les bits de configuration E/S (IODIRA/B). Les données pour chaque entrée ou sortie sont conservées dans le registre d'entrée ou de sortie correspondant. La polarité du registre de port d'entrée peut être inversée grâce au registre d'inversion de polarité. Tous les registres peuvent être lus par le système maître. Une interface I2C haute vitesse assure la communication entre le système maître et l'expander, ce dernier contient aussi un bouton Reset ainsi que trois broches d'adresses (A0A1A2) permettant de connecter jusqu'à 8 appareils à l'expander (dans notre cas les broches sont fixés à l'adresse 000), deux broches INTA et INTB d'interruption qui peuvent être configurés pour fonctionner indépendamment ou conjointement.

10.1 Principes

L'interface MIDI est l'interface qui permet au synthétiseur PULSAR d'être commandé par un clavier MIDI. Le terme MIDI est l'acronyme de Musical Instrument Digital Interface, c'est un protocole de communication conçu en 1983 qui permet de piloter un dispositif musical à travers un appareil contrôleur.

Le protocole MIDI ne transmet pas de signal audio, mais uniquement des messages de commandes comme une partition musicale. Les informations sont codées de manière numérique et transmises à une vitesse de 31250 bauds sur un bus MIDI (qui est un bus

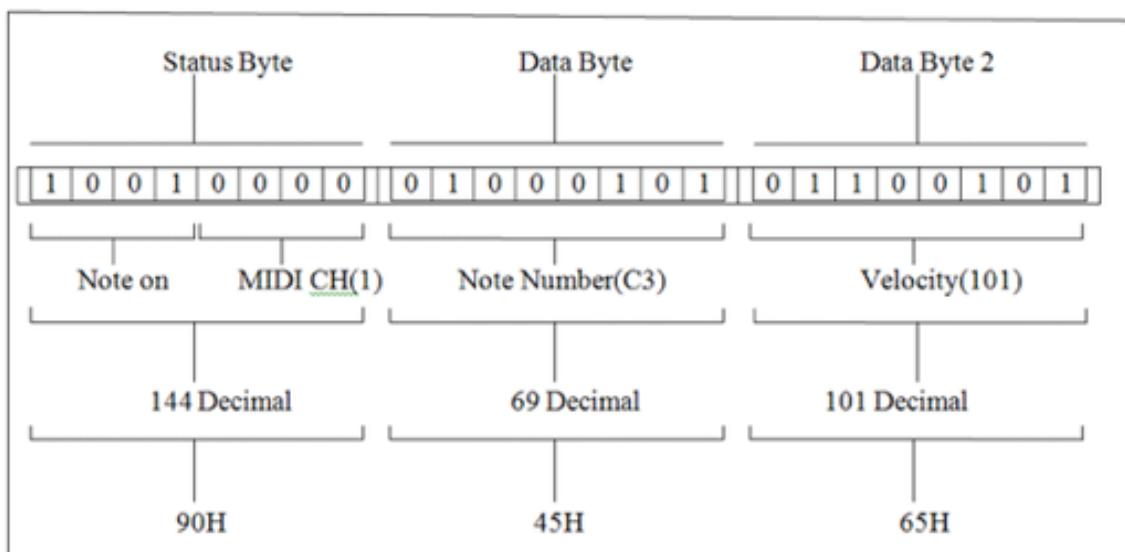


Figure 10.1: Exemple de message MIDI

série asynchrone). Chaque octet transmis est précédé d'un bit de start et suivi d'un bit de stop, soit au total 10 bits transmis (figure 10.1). La plupart des messages contiennent un identifiant de canal, cela donne la possibilité de multiplexer 16 canaux pour commander plusieurs instruments distinctement à travers un même câble.

MIDI number	Note name	Keyboard	Frequency
21	A0		27.500
22	B0		30.868
23	C1		32.703
24	D1		36.708
25	E1		41.203
26	F1		43.654
27	G1		48.999
28	A1		55.000
29	B1		61.735
30	C2		65.406
31	D2		73.416
32	E2		82.407
33	F2		87.307
34	G2		97.999
35	A2		110.00
36	B2		123.47
37	C3		130.81
38	D3		146.83
39	E3		164.81
40	F3		174.61
41	G3		196.00
42	A3		220.00
43	B3		246.94
44	C4		261.63
45	D4		293.67
46	E4		329.63
47	F4		349.23
48	G4		392.00
49	A4		440.00
50	B4		493.88
51	C5		523.25
52	D5		587.33
53	E5		659.26
54	F5		698.46
55	G5		783.99
56	A5		880.00
57	B5		987.77
58	C6		1046.5
59	D6		1174.7
60	E6		1318.5
61	F6		1396.9
62	G6		1568.0
63	A6		1760.0
64	B6		1975.5
65	C7		2093.0
66	D7		2349.3
67	E7		2637.0
68	F7		2793.0
69	G7		3136.0
70	A7		3520.0
71	B7		3951.1
72	C8		4186.0
73			J. Wolfe, UNSW
74			
75			
76			
77			
78			
79			
80			
81			
82			
83			
84			
85			
86			
87			
88			
89			
90			
91			
92			
93			
94			
95			
96			
97			
98			
99			
100			
101			
102			
103			
104			
105			
106			
107			
108			

Figure 10.2: Codes MIDI pour les numéros de notes du clavier.

10.2 L'interface Midi du PULSAR

Pour l'instant, l'interface MIDI du PULSAR ne gère que les messages "Note On" et "Note Off".

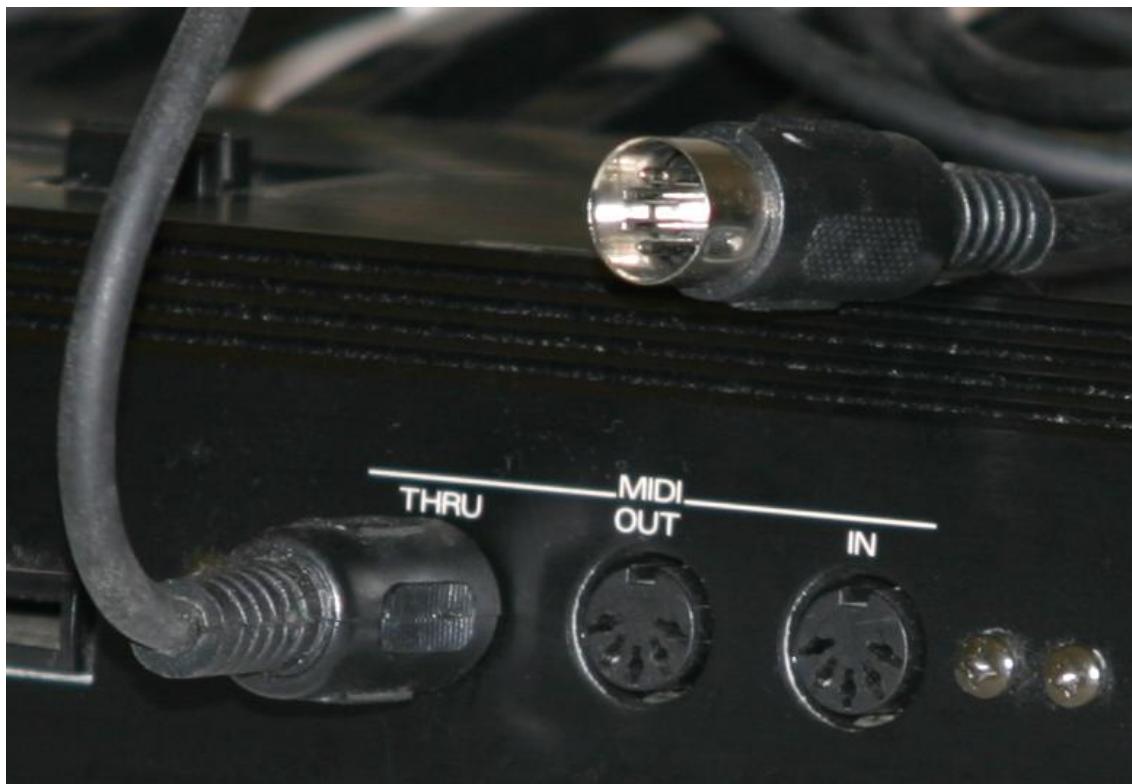


Figure 10.3: Exemple de connectique MIDI : IN = entrée depuis un clavier, OUT = sortie vers un instrument, THRU = répéteur de l'entrée IN pour éviter l'atténuation du signal

Notre liaison Midi exploite une liaison série "maison" de type "BitBang" car la contrainte de fréquence de communication (31250 bit/s) ne nous permet pas d'utiliser la liaison série "hardware" de la Raspberry PI 3, qui ne supporte que les débits standards pour les UART (à savoir 9600, 14400, 19200, 38400, 57600, 115200). Une liaison série "bitbang" est une liaison émulée intégralement de manière logicielle. Elle est plus coûteuse en temps de calcul, mais elle offre plus de souplesse ici.

Cette liaison série "BitBang" exploite les fonctions "gpioSerial" de la librairie Piggpio que nous avons légèrement modifiées, ainsi à l'avenir il est important d'utiliser les fichiers source "pigpio.c" modifié, cette liaison ne nécessitant par ailleurs qu'un port GPIO et l'alimentation habituelle 3.3V.

La liaison Midi utilise un câble midi transportant du 5V TTL, on a alors besoin d'un montage pour isoler cette tension des 3.3V des GPIO de la carte Raspberry PI 3 avec des optocoupleurs 6N138. On a donc le schéma du montage de la carte MIDI IN du synthé PULSAR illustré figure 10.4

Les messages MIDI arrivent par octets comme sur la figure 10.1, une interruption dans le code source pigpio.c intervient alors pour appeler la fonction "midi_byte_received_handler" qui stocke l'octet dans une variable appelée "octet" puis c'est cette même fonction qui s'occupera de décrypter le message en trois informations : le type de command control (note on ou note off), la note concerné et la vitesse de la note (= la force avec laquelle on appuie sur la touche du clavier). Cette lecture d'information se fait par l'intermédiaire d'une machine à état fonctionnant de la manière suivante :

- On va tout d'abord attendre la détection d'un premier octet "control change" reconnue

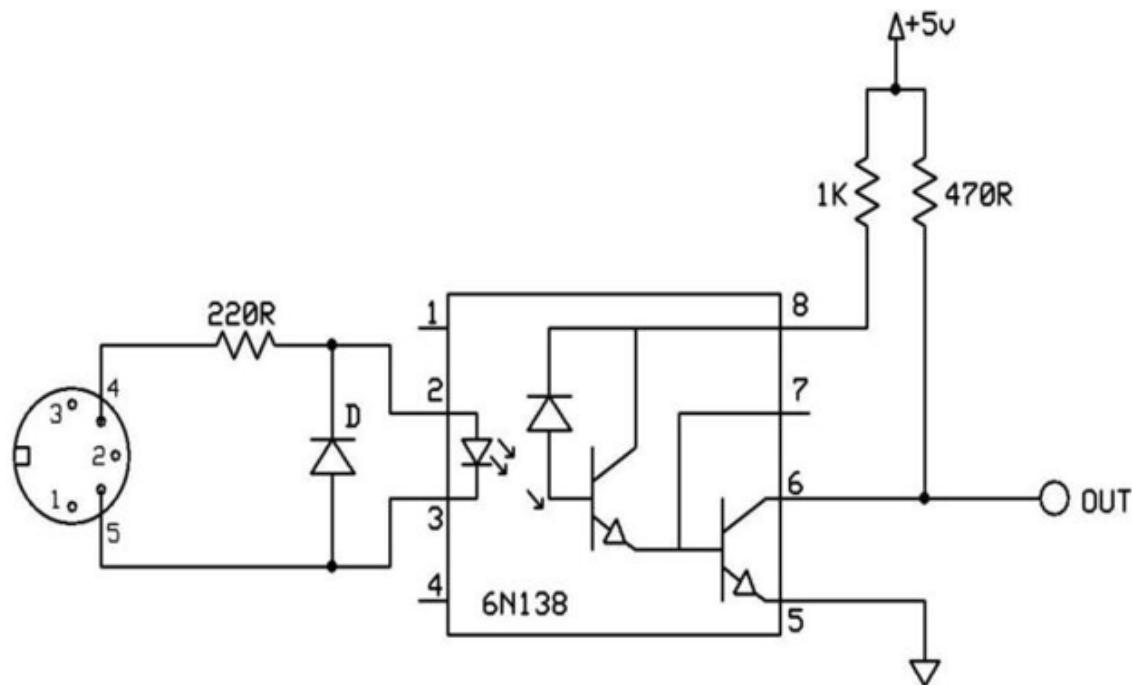


Figure 10.4: Montage de la carte Midi IN

(Note On ou Note Off)

- on passe ensuite dans un état d'attente de l'octet indiquant le numéro de la note concernée
- lorsque ce dernier est reçu on passe dans un état d'attente de l'octet de la vitesse concernée.

Les trois états possibles de la machine à état sont donc :

1. MIDI STATE MACHINE IDLE : attente d'un message control change
2. MIDI STATE MACHINE WAITING NOTE : attente d'une Note
3. MIDI STATE MACHINE WAITING VELOCITE : attente d'une Vélocité.

```
// fonction appelee lorsque un octet est disponible
// sur le port serie
void midi_byte_received_handler(uint32_t octet){

    if (octet == NOTE_ON || octet == NOTE_OFF){
        midi_msg_type = octet;
        midi_machine_state = WAITING_NOTE;
    }

    else if (midi_machine_state == WAITING_NOTE ){

        midi_msg_note = octet;
        midi_machine_state = WAITING_VELOCITE;
    }

    else if (midi_machine_state == WAITING_VELOCITE){

        midi_msg_veloce = octet;
        midi_machine_state = IDLE_MIDI;

        if (midi_msg_type == NOTE_ON){
            midi_note.note = midi_msg_note;
            midi_note.veloce = midi_msg_veloce;
            note_on_handler();
        }
        else if (midi_msg_type == NOTE_OFF){
            midi_note.note = midi_msg_note;
            midi_note.veloce = midi_msg_veloce;
            note_off_handler();
        }
    }
}
```

Figure 10.5: Code du handler de messages MIDI.

```
void note_on_handler(){

    // changement de l'état de la state machine ADSR:
    state_machine_vca->t = 0;
    state_machine_vca->tmp_exp=1.0;
    state_machine_vca->machine_state=ATTACK;
    // envoie des tensions analogiques sur les
    // VCF via le DAC 4822
    write_DAC_note(midi_note.note);
}
```

Figure 10.6: Code du gestionnaire de messages MIDI "Note On" : on modifie l'état courant de la machine à état qui gère les enveloppes ADSR.

Le projet global du synthétiseur hybride Pulsar possédera un module supplémentaire en sortie afin d'enrichir le son. Ce dernier module aura deux fonctions principales, un module d'effet pour le son, pour lui donner une "couleur" propre à la distorsion utilisant la technologie des lampes, ainsi qu'une deuxième fonction purement amplificatrice. Afin d'avoir une utilisation agréable, nous rendons ce module paramétrable, notamment sur le volume de sortie et le taux de distorsion du signal via l'interface physique (potard) ou numérique, le choix étant laissé à l'utilisateur. Ce module est étudié pour pouvoir être utilisé en sortie du synthétiseur, ou bien de manière indépendante en tant que préamplificateur de guitare électrique.

Le projet Synthétiseur hybride analogique/numérique présente un choix global important, le Signal Path doit être analogique. Ce type de circuit analogique est notamment recherché par les audiophiles ; il permet une restitution du son souvent jugée 'plus authentique' que le numérique. En effet, nous pouvons comprendre qu'un échantillonnage cause forcément une perte de l'information de départ lors du passage de valeurs continues à des valeurs discrètes. D'où notre intérêt de garder un signal analogique et en privilégiant l'Analog Path.

Malgré l'apparition en 1946 des amplificateurs à transistors, et plus récemment des amplificateurs à modélisation, les lampes sont toujours présentes de nos jours !

L'invention de la lampe à amplification remonte à 1904 notamment grâce aux travaux de Ambroise Felming et Lee De Forest. Le principe est le suivant : Dans une enceinte sous vide (d'où l'ampoule) un filament est chauffé (la cathode ou la source) et émet des électrons circulant jusqu'à la une tôle de métal enroulée, l'anode (ou le drain). Entre les deux, De Forest rajoute un fil tournant autour de la cathode sans être en contact avec les deux éléments : la grille est née, permettant une première amplification du signal.

Pour ce qui est des avantages, les guitaristes le savent bien, contrairement au transistor, l'ampli à lampe sature plus doucement lorsqu'on augmente le gain, là où la saturation des

transistors est plus tranchée. Le son est donc différent, car les harmoniques ajoutées par le composant ne sont pas les mêmes.

Ces différences au niveaux des harmoniques générées seront abordées plus loin.

Cette partie suivra donc le plan suivant :

1. Cahier des charges
2. Architecture du préamplificateur
3. Ecart au cahier des charges initial et améliorations du prototype

11.1 Cahier des charges

En regard à l'ensemble des caractéristiques des autres modules du synthétiseur, et aux caractéristiques déjà évoquées en introduction de cette partie, dressons un cahier des charges fonctionnel pour notre préamplificateur à lampe. Notre préamplificateur doit :

- Permettre une saturation du signal par des triodes, particulièrement intéressante dans le cadre d'un projet audio.
Nous décidons d'étudier un système exploitant le Nutube que nous étudions plus tard, composant constitué de deux triodes, très utiles pour amplifier le signal jusqu'à sa saturation, la saturation des triodes ajoutant des harmoniques particulières au son. La partie "triode" est basée sur un nouveau composant fabriqué par Korg, le "Nutube", qui intègre deux triodes sur un chip de petite taille, et qui la nouveauté appréciable l'autorise des tensions de polarisations "raisonnables", autour de quelques dizaines de Volts (la ou des triodes, pentodes classiques exigent plusieurs centaines de Volts) tous les paramètres (gain, couleur, distorsion, etc) sont programmables/mémorisables par la carte RPi3 ; l'implémentation utilise des potentiomètres numériques programmables de type Microchip MCP4251 ou bien des modules VCA le module est débrayable via l'interface graphique de la RPi3.
- Respecter un signal Path analogique : maximiser les traitements analogiques du signal et minimiser un traitement numérique du signal afin de garder la couleur, les imperfections du signal amplifié.
- Être adapté à un signal venant du synthétiseur ou d'une guitare électrique. Le préamplificateur a donc deux fonctions distinctes pour chacun des cas d'utilisation : Pour l'utilisation en tant que préamplificateur de guitare à saturation, il est utilisé dans une chaîne classique d'amplification qui comprend un préamplificateur, un amplificateur de puissance et un haut-parleur en fin de chaîne. De plus, concernant l'utilisation en tant que module d'effet de saturation pour le synthétiseur PULSAR intercalable dans une chaîne d'effets audio.
- Utiliser la même alimentation que le synthétiseur afin d'éviter d'alourdir le montage global et de multiplier les voies d'alimentation.
- Ramener le niveau du signal d'entrée à quelques volts en sortie (ordre de grandeur de 1V) si nécessaire.

Pour un pré-amplificateur de guitare, un gain situé entre 20 et 100 serait acceptable, car une guitare délivre un signal d'environ 100 mV. En considérant le module comme une pédale d'effet pour le synthétiseur hybride, il n'y a pas besoin de gain supplémentaire étant donné que le niveau d'entrée est déjà de quelques volts, ce qui suffit amplement à un amplificateur de puissance.

11.2 Architecture du préamplificateur

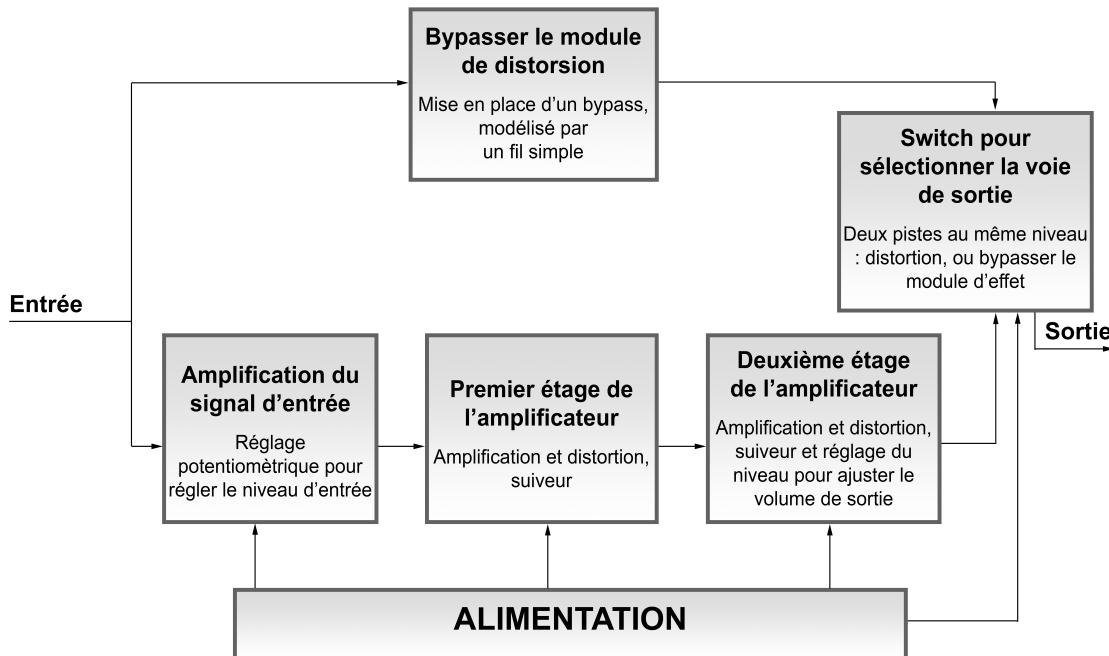


Figure 11.1: Schéma bloc du préamplificateur à tubes

L'alimentation du synthétiseur nous fournit les tensions -15V, +15V, 0V et 5V.

Le Nutube étant un ensemble de deux triodes miniaturisées, nous les utilisons telles deux amplificateurs de Classe A en cascade. T1 et T2 sont donc tous deux des circuits à émetteur commun construits autour d'une triode du Nutube. Nous reviendront sur leur fonctionnement dans la partie suivante.

Notre préamplificateur doit amplifier un signal issu d'une guitare électrique, d'une amplitude typique de 100 mV. Bien évidemment cette valeur peut varier selon les micros, pouvant aller jusqu'à quelques volts si l'on connecte un synthétiseur. Elle n'est pas tout le temps suffisante pour atteindre le seuil de saturation des triodes, d'où le choix d'un AOP en non-inverseur afin d'obtenir un gain de 26 préalable. Il est donc important de pouvoir régler le niveau du signal entrant si nécessaire pour ne pas faire saturer cet AOP, un potentiomètre d'entrée est à prévoir pour diviser le niveau d'entrée suffisamment..

Le potentiomètre numérique d'entrée permet un contrôle du "pré-gain" via le bus SPI d'une Raspberry PI, et donc via une interface numérique utilisateur (écran tactile). Il permet d'ajuster le niveau du signal entrant dans la première triode, et donc de jouer sur le niveau de saturation. Celui en sortie de la deuxième triode permet d'ajuster le volume de sortie (aucun effet sur la saturation).

Si le montage est utilisé comme effet pour le synthétiseur, il doit pouvoir être contourné à tout moment sans couper le signal, d'où la présence d'un switch (bypass) en sortie que l'on peut également contrôler via le Raspberry PI.

Description du schéma électrique détaillé

Etudions pas à pas le chemin de notre signal dans le schéma décrit sur la figure N :

Première fonction En entrée du montage, plaçons une capacité de liaison afin de supprimer tout offset possible. et ne garder que la partie AC du signal. La valeur de $1 \mu F$ est choisie afin que l'effet passe haut ne soit pas perceptible au niveau du son $f_0 < 20Hz$. L'ordre de grandeur de résistances reliées à la masse suivant ces capacités est de $100k\Omega$ ou $10k\Omega$ on a donc ces deux ordres de grandeur :

$$\text{Pour } 100k\Omega : f_0 = 1/(2\pi * 100k * 1\mu) = 1,59Hz < 20Hz$$

$$\text{Pour } 10k\Omega : f_0 = 1/(2\pi * 100k * 1\mu) = 15,9Hz < 20Hz$$

Cela est valable pour l'ensemble des capacités de liaison de notre montage, à savoir : $C_1 C_2 C_3 C_4 C_7 C_8 C_9 C_{10}$ et C_{11} . Seule C_5 doit être plus grande, à cause du potentiomètre.

Deuxième fonction La voie du dessus (bypass), est directement en sortie, avec le switch en sortie. Cela permet à l'utilisateur de choisir à tout moment de d'annuler la saturation et de restituer un signal identique à celui en entrée, requis pour une pédale d'effet.

Troisième fonction Pour la voie du bas, le potentiomètre numérique permet de diviser le niveau du signal d'entrée. Si le signal est déjà élevé en entrée ($>1V$) il vaut mieux éviter de faire saturer l'AOP non-inverseur qui suit et dont le gain est de 26. C'est ce potentiomètre qui permet à l'utilisateur de choisir le niveau de distorsion (réglage utilisateur "pré-gain")

Quatrième fonction AOP non inverseur ont deux rôles :

- Isoler la triode de l'entrée, pour éviter les problèmes d'impédances
- Donner un gain de 26 au signal d'entrée afin qu'il ait une amplitude suffisante pour arriver au moins à la limite de saturation de la triode 1. Le gain d'un AOP non-inverseur : $1 + \frac{R_5}{R_6} = 26$.

Cinquième fonction Première triode du Nutube :

Le préamplificateur doit apporter de la distorsion au signal. Cela arrive lors de la saturation du composant d'amplification tel qu'un transistor, ou bien une triode. L'écrêtage du signal engendre alors l'apparition de nouveaux harmoniques dans le spectre fréquentiel du signal, ce qui rend évidemment le timbre du son différent une fois restitué par les haut-parleurs.

Polarisation de la première triode : Pour faciliter l'étude on assimilera ici le [-15V + 15V] en [0V-30V]. Avant tout, le courant de filament doit être proche de 20 mA (valeur constructeur à respecter strictement afin de ne pas endommager le composant) d'où les résistances $R_1 R_2$ et R_3 pour diviser le courant.

Afin d'évaluer l'amplification, et le seuil de saturation d'une triode nous nous intéressons à sa caractéristique: avec le point de polarisation correspondant à 2,5V DC (1) sur la grille E_{G0} (imposée par le pont diviseur $R_9 R_{10}$) et une résistance d'anode de $330k\Omega$ et $V_{CC} = 30V$ (droite de charge du haut sur la caractéristique). En régime dynamique le signal impose l'oscillation de la tension E_G autour de ce point de fonctionnement.

Ainsi pour des variations de E_G plus de 2,5V AC (2), nous arrivons dans le domaine

de saturation de la triode, où les courbes sont coudées, pour la saturation haute et 0 pour la saturation basse.

Remarques :

- (1) En modifiant E_{G0} , la saturation peut être rendue plus ou moins “symétrique”.
- (2) Expérimentalement il faut un peu plus pour arriver à une saturation significative : 3V voir 5V.

En regardant la caractéristique, le courant peut donc théoriquement monter à environ $I_{Amax} = 82\mu A$:

$$V_{Smin} = 30 - R_A * I_{Amax} = 2,94V$$

$$V_{Smax} = 30V$$

Ce qui explique pourquoi l'amplitude crête à crête de notre signal en sortie est limitée à 27,94 V, donc quasiment 14V d'amplitude :

$$V_{max(Anode)} = 14V$$

Heureusement nous n'avons pas besoin d'autant de tension en sortie d'un préamplificateur, quelques volt suffisent.

En régime linéaire le gain typique d'une lampe est dit de 5, d'après la datasheet. En calculant :

$$G = \frac{V_{outmax}}{V_{inmax(nonsature)}} = \frac{14}{2.5} = 5,6.$$

En prenant en compte l'observation expérimentale (2) on se rapproche bien de 5. Supposons une entrée guitare 100 mV max d'amplitude sans aucune réduction par le potentiomètre d'entrée : du fait de l'amplification du signal par 26 par l'AOP précédent, il attaque la grille de la triode avec $E_G = E_{G0} \pm 2,6V$.

Cela permet au signal d'atteindre la saturation haute et basse au niveau de ses crêtes. Cependant, il faut noter que la plupart du temps, le signal de la guitare sera inférieur à sa valeur maximale, en dessous de 60 mV de la guitare, la lampe ne sature plus. Pour un jeu “normal” du guitariste, une faible proportion du signal est saturée ce qui s'entend peu. Il faut en saturer d'avantage !

Sixième fonction Utilisation d'un suiveur en sortie de la première triode, cela évite les problèmes d'impédance entre les deux lampes : cela permet d'utiliser le gain maximum de chaque triode de manière indépendante.

Septième fonction Deuxième triode du Nutube, polarisée de la même manière que la première :

Le signal ayant subit un gain de 26, puis, de 5 grâce à la triode 1 (on suppose que les potentiomètres précédents sont au maximum et n'atténuent pas le signal). Cette fois son amplitude est majoritairement supérieure aux 2,5V nécessaire à la saturation.

$$V_E = 2,5 / (26 * 5) = 0,019V$$

Un signal de 19 mV d'amplitude en entrée du système suffit à faire saturer cette triode. La plupart des notes jouées par le guitariste dépassent cette amplitude et pourront donc être saturées (bien évidemment le taux de saturation dépend encore une fois de l'intensité de chaque note).

A ce stade l'amplitude est maintenant réduite volontairement (pont de résistances R_4 et R_{d1} donnant un gain de 0,3), pour ne pas faire saturer le buffer de sortie (AOP) et atteindre un niveau de sortie demandé de quelques volts. Amplitude maximale en sortie :

$$V_{max(Anode)} * 0,3 = 14 * 0,3 = 4,2V$$

Huitième fonction Nous plaçons un suiveur afin d'éviter d'isoler la deuxième triode de la sortie.

Neuvième fonction Deuxième potentiomètre numérique afin de régler cette fois le niveau du signal de sortie, et donc l'équivalent du réglage utilisateur "Volume".

Dixième fonction Mise en place du switch qui permet d'alterner entre l'amplificateur et la voie de bypass. On place des capacités de liaison C_7 et C_8 afin d'éviter une différence de niveaux entre les deux voies (craquement lors du switching).

Remarque sur les triodes :

Pourquoi avoir choisi des triodes ?

Elles sont aujourd'hui encore très utilisées dans l'audio, notamment du fait que leur réseau de caractéristique courant/tension leur confère des propriétés différentes des transistors :

1. La saturation est beaucoup moins abrupte que celle des transistors.
2. Pour un son de guitare, les harmoniques apportées par la saturation donnent une sensation auditive généralement plus proche du blues/rock/funk que pour un transistor, plus proche du metal (à fortiori un transistor bipolaire).
3. La caractéristique d'un tube en fonctionnement linéaire est généralement moins proche d'une droite que celle d'un MOS par exemple. Cela confère donc une dynamique particulière au son, moins linéaire.

Simulations réalisées avec LT-spice confrontant des 12AX7 et un MOSFET :

Nous menons des simulations LTSpices afin de visualiser l'impact sur les harmoniques générées. Cependant, ces dernières sont limitées à la fidélité des modèles de composants utilisés.

Remarque sur les potentiomètres numériques :

Pour chaque potentiomètre numérique l'utilisateur peut également choisir d'utiliser un potentiomètre analogique à la place, cela est réglable via deux commutateurs leviers (voir câblage Eagle).

Remarques sur le choix des composants au formats DIP :

Utilisation d'un AOP de type NE5534 qui est un amplificateur opérationnel audio (faible bruit et faible distorsion). Switch MAX4541 qui permet de faire une sélection simple entre deux pistes via une entrée numérique. Les potentiomètres numériques sont des MCP41100 de 100k max utilisable via un Raspberry Pi commandé par un bus SPI.

Potentiomètre numérique : MCP41100 Contrôle SPI: Chip Select (CS)

Tout message SPI adressé au potentiomètre commence par la mise à 0 du CS puis termine par l'envoie d'un 1 sur ce dernier. La datasheet donne les informations suivantes pour régler la résistance: Il faut envoyer au Data Input un premier message de un octet indiquant la commande à effectuer.

Pour la commande "write data" , il faut envoyer le nombre binaire xx01xx01 dans lequel la valeur des x n'a aucune importance. Si on remplace les x par des 0, on envoie donc le nombre binaire 00010001, qui correspond au nombre hexadécimal 11, ou tout simplement au nombre décimal 17.

Il faut ensuite envoyer un deuxième message d'un octet: un nombre entre 0 et 255 qui indique à quelle valeur la résistance doit être réglée (0 pour la valeur minimale, 255 pour la valeur maximale).

Horloge: Comme son nom l'indique, horloge du microcontrôleur utilisé

Switch : MAX 4544 Cf : schéma issude la datasheet

Très simple d'utilisation : l'état de IN permet de permutez entre NO et NC : (voir 11.10)

Etat de IN	Etat de OUT
0	NC
1	NO

11.3 Ecart au cahier des charges et améliorations du prototype

Le projet est relativement bien avancé : la partie théorique et de la réalisation du préamplificateur est finalisée. Le prototype pourrait être testé une fois sur PCB afin d'industrialiser le préamplificateur. Autrement les performances annoncées dans le cahier des charges sont respectées.

Nous avons décidé de le réaliser de sorte à l'utiliser comme un module à part, ou comme une partie du synthétiseur Pulsar ce qui permet une plus grande liberté dans l'utilisation.

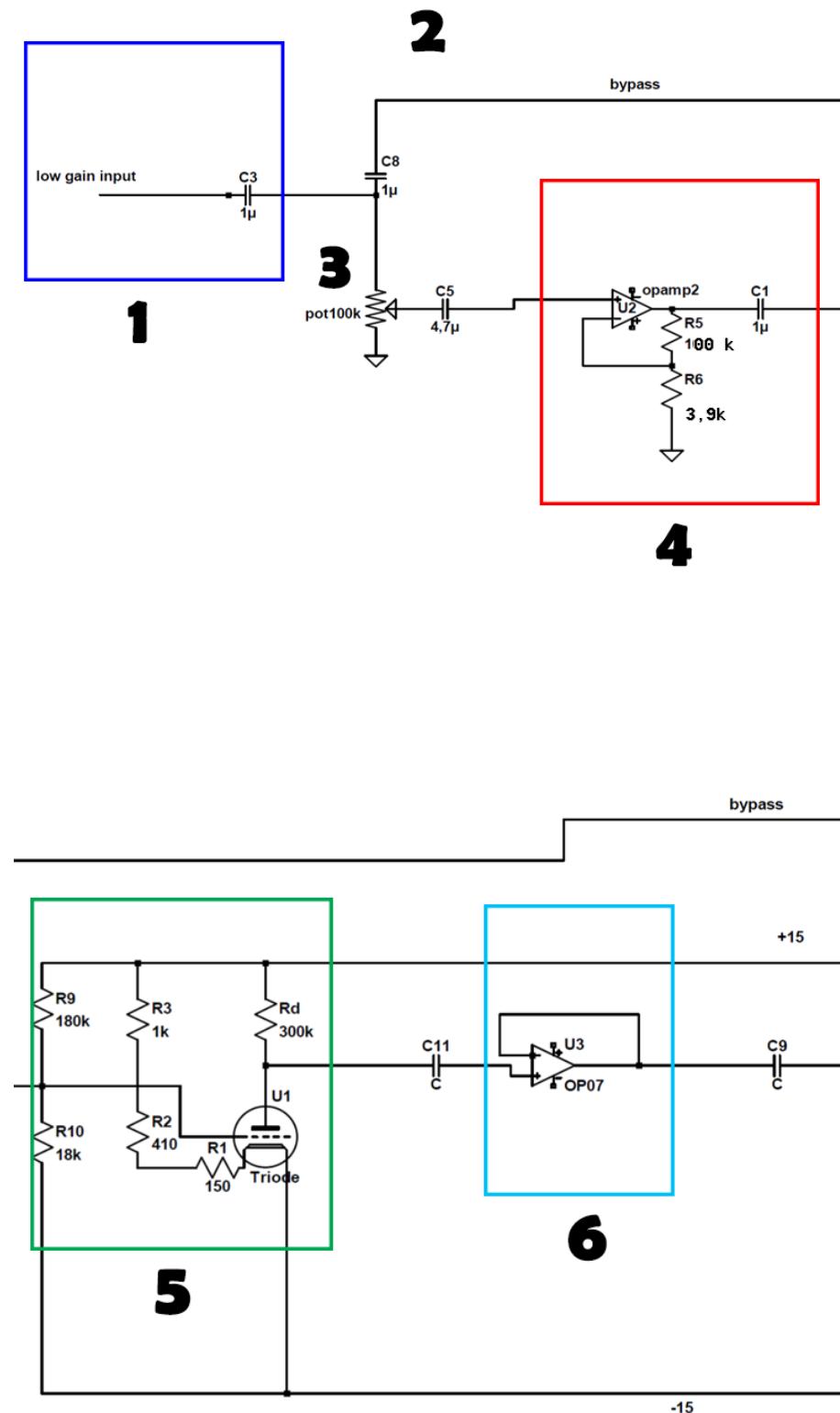


Figure 11.2: Première partie du montage : étude pas à pas du préamplificateur

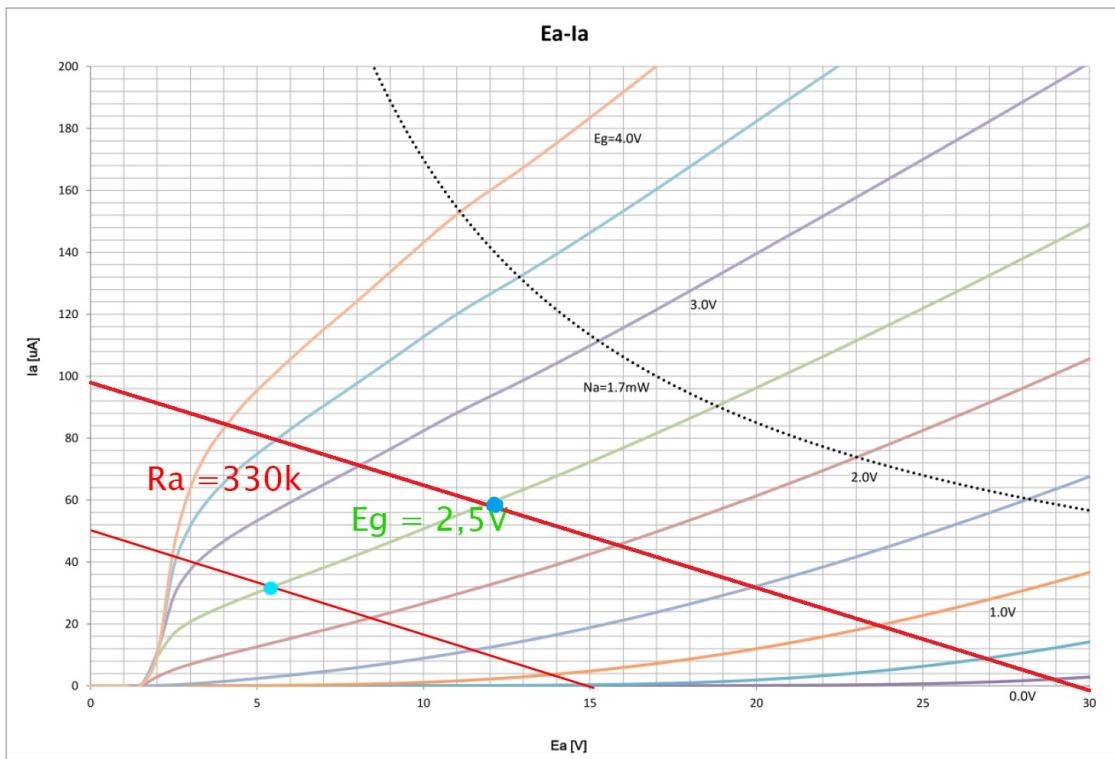


Figure 11.3: Caractéristique d'un triode, résistance d'anode 330k

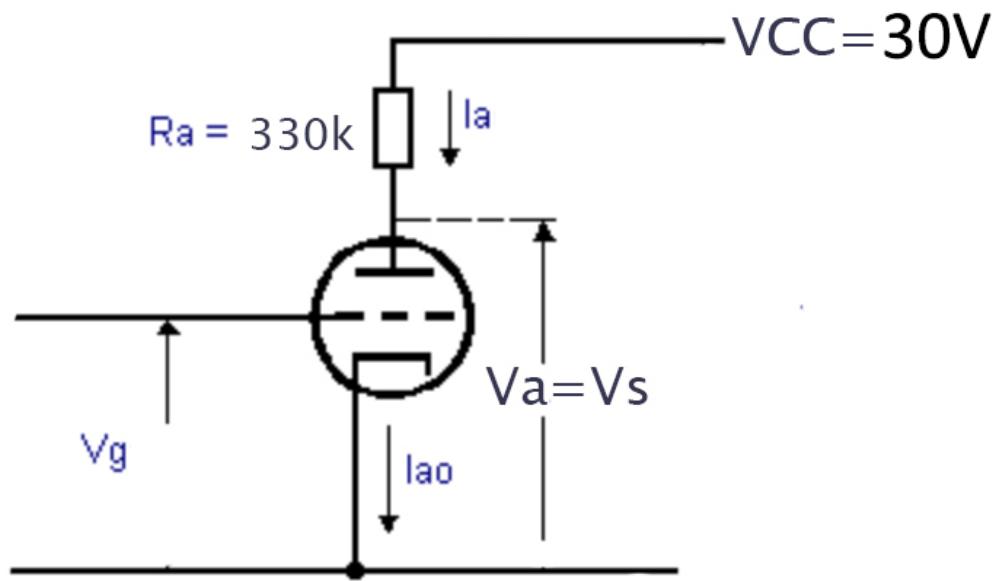


Figure 11.4: schéma du circuit de polarisation

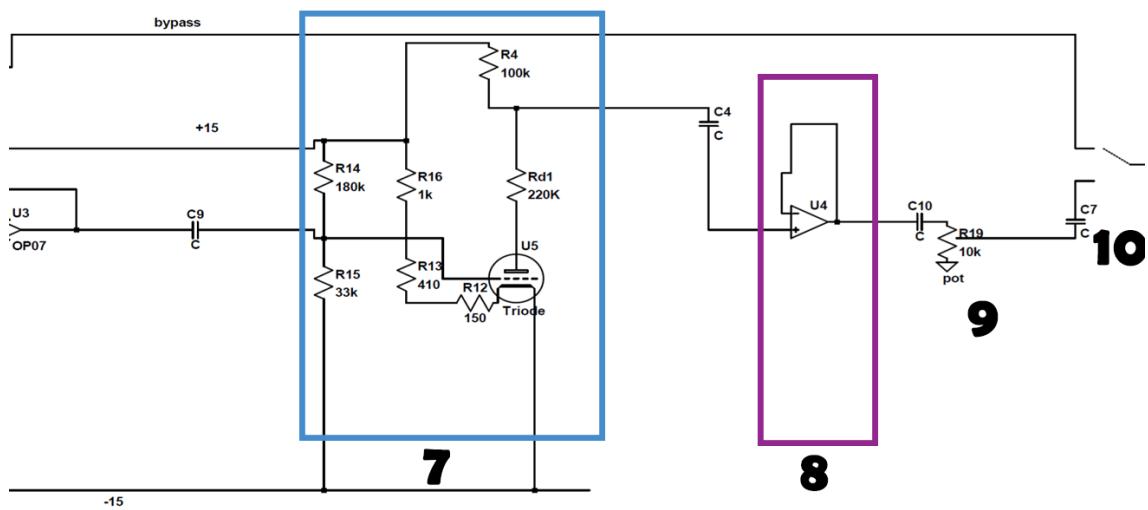


Figure 11.5: Deuxième partie du montage : étude pas à pas du préamplificateur

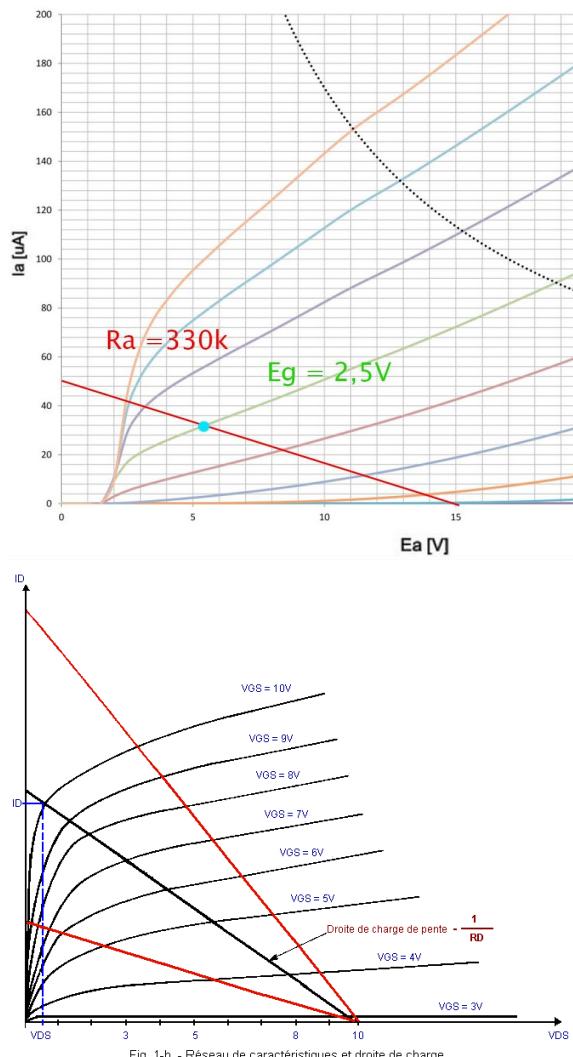


Fig. 11.6 - Réseau de caractéristiques et droite de charge.

Figure 11.6: Comparaison des caractéristiques d'une triode du Nutube (en haut) et d'un MOSFET(en bas)

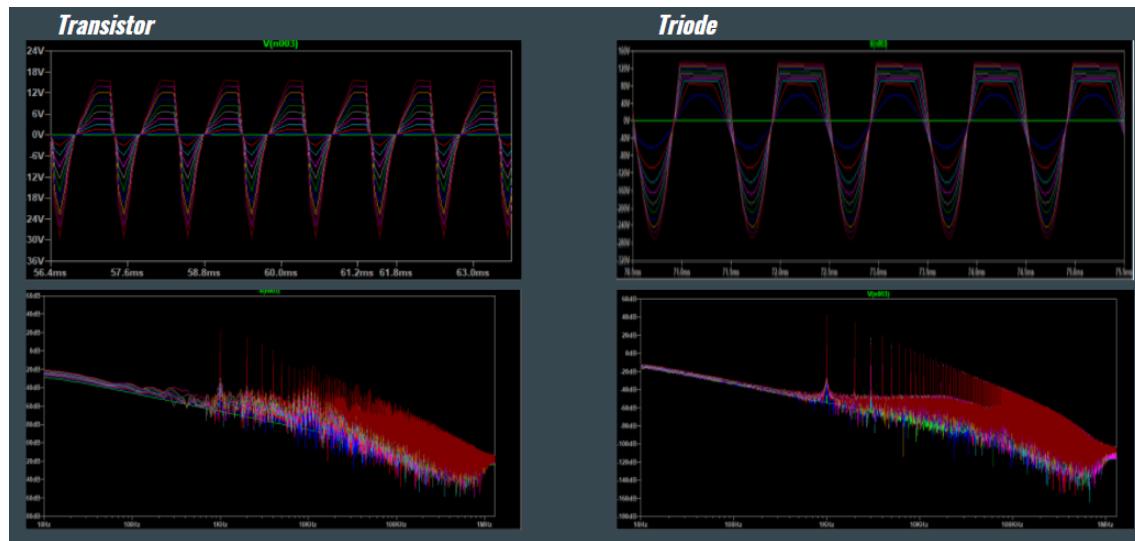


Figure 11.7: Simulation du modèle : comparaison transsistor et triode

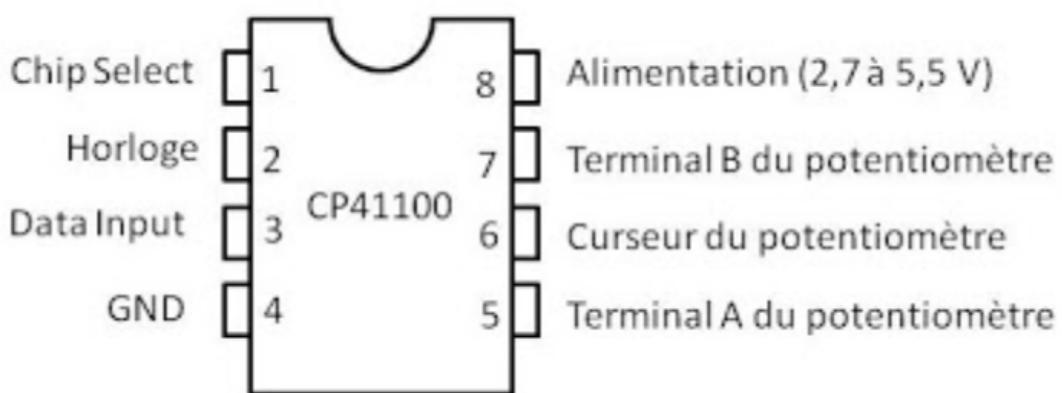


Figure 11.8: Extrait de la datasheet du potentiomètre numérique : MCP41100

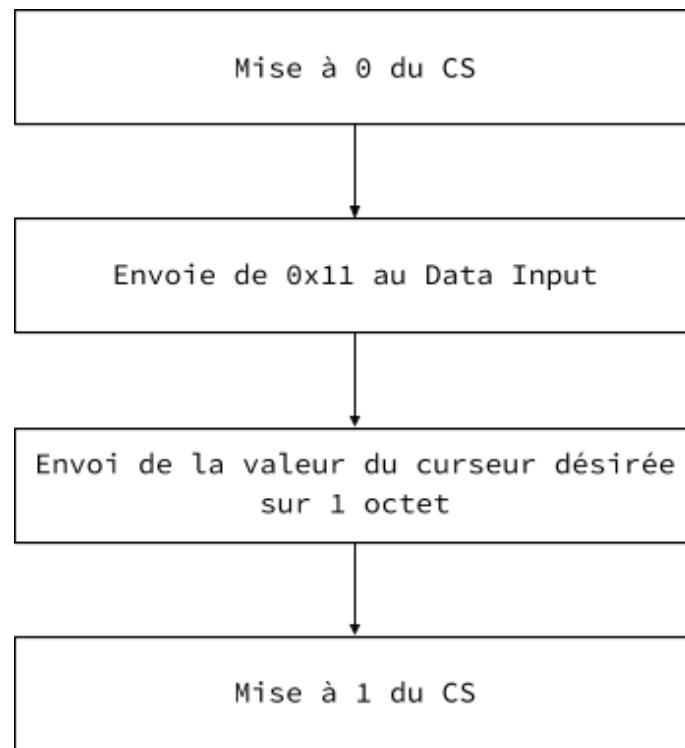


Figure 11.9: Utilisation du Potentiomètre Numérique

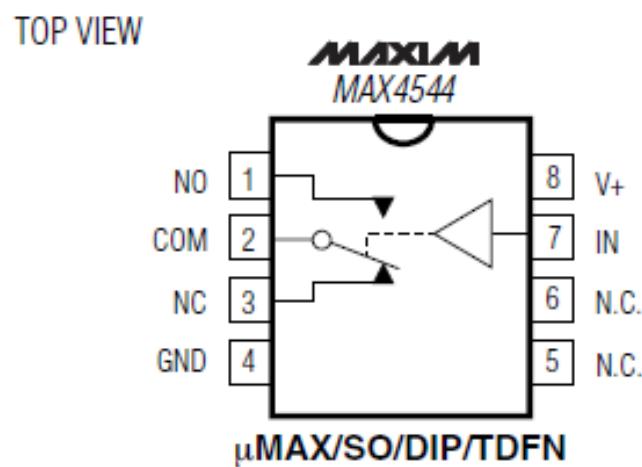


Figure 11.10: Extrait de la datasheet du switch : Max 4544

Au coeur du Pulsar réside une forte volonté de créer le produit le plus évolutif possible. Le module “Synthétiseur” principal doit pouvoir communiquer avec un maximum d’éléments extérieurs afin de permettre aux artistes, comme aux bidouilleurs de laisser leur créativité s’exprimer au maximum.

Le rack d’effets développé au cours de cette année fait ainsi office de modèle pour la création de toutes sortes d’effets additionnels à brancher en sortie du synthétiseur. Seule subtilité, les modules présents dans le rack doivent aussi pouvoir communiquer avec la Carte Raspberry Pi 3, centre névralgique de l’interface homme machine.

Le delay consiste en un retard d’une durée fixe appliquée à un signal envoyé en entrée (ici produit par le synthétiseur). Le delay est au centre de la création de la réverbération. En effet, la réverbération s’obtient en mélangeant le signal retardé à l’aide du delay au signal initial. Lorsque le rythme de cette réinjection ne permet plus distinguer le son initial, du son “rebondi”, on dit qu’il y’a une réverbération.

L’un des exemples les plus communs de réverbération s’observe dans l’acoustique des salles. Le physicien Wallace Clement Sabine est le premier à avoir entrepris des études sur la réverbération, avec des mesures dans des salles existantes et l’établissement d’un modèle basé sur les lois de propagation du son.

Pour imiter cet effet, il est possible de réaliser plusieurs répétitions du son initial, en réinjectant notre signal de sortie sur l’entrée de notre module de retard et en veillant à ce que l’amplitude de notre signal réinjecté soit inférieure à celle du signal initial. Lorsque on arrive à distinguer le son initial, du son qui revient retardé, c’est que le retard est suffisamment grand (plusieurs dizaines de millisecondes). On parle ici d’écho.

La commande de ces deux effets via notre interface tactile doit nous permettre d’enrichir le son de notre synthétiseur. L’objectif final du projet est de pouvoir invoquer des presets préalablement créés depuis l’interface graphique. Cette fonctionnalité n’a finalement pas

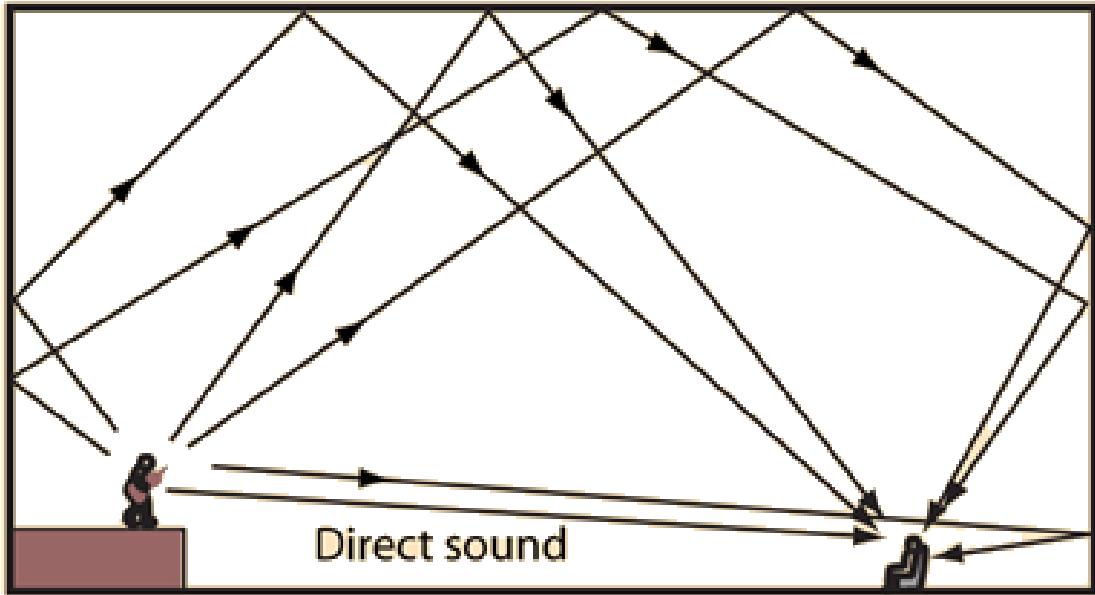


Figure 12.1: Exemple de réverbération à deux dimensions. Le retard est ici naturellement produit par la réflexion du son sur les murs de la salle.

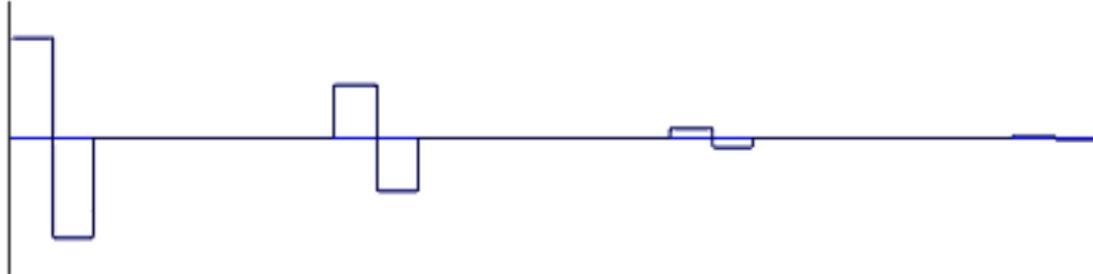


Figure 12.2: Exemple d'écho avec amplitude décroissante.

été implémenté et seul la durée du retard est modifiable via une commande numérique séparée de l'interface graphique.

12.1 Fonctionnement Général

Le module de Réverbération/Delay est conçu pour être connecté en bout de chaîne, à la sortie audio du synthétiseur. Ce dernier se compose de deux éléments :

- Une carte électronique comportant les composants analogiques permettant la réalisation de l'effet. Cette carte est connectée à la sortie jack du synthétiseur et possède elle-même une sortie jack permettant de la cascader avec un autre effet ou à connecter à la sortie son.
- Une carte de développement STM32 permettant la communication avec la carte Raspberry Pi 3 ainsi que la génération et le contrôle des signaux nécessaires au

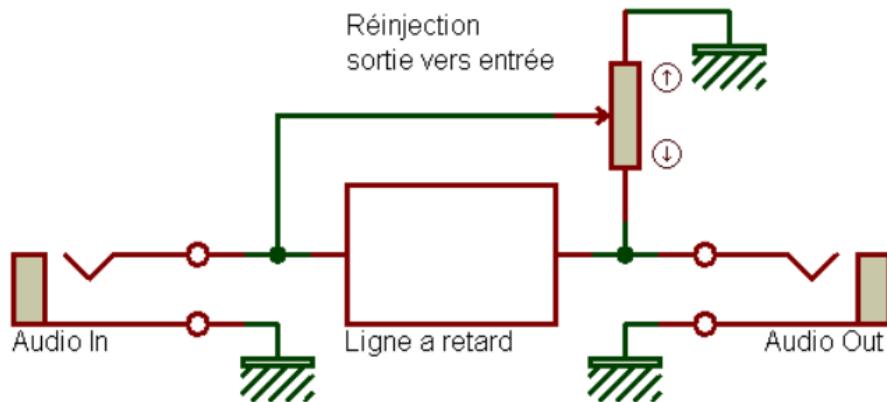


Figure 12.3: Schéma d'implémentation d'un delay basé sur une ligne à retard (delay).

fonctionnement de la carte analogique avec qui elle communique via ses GPIOs.

La carte STM32 permettant de faire bien plus que générer les signaux permettant le contrôle de notre carte analogique grâce à sa puissance, nous utilisons aussi cet élément du module à d'autres fin. En effet, la Raspberry devant gérer l'interface graphique ainsi que plusieurs autres tâches gourmandes en ressources, nous utilisons aussi la carte STM32 pour la partie "Synthèse par Wavetable". La carte électronique nous permet ainsi de synthétiser des formes d'ondes numériques utilisables en tandem avec nos formes d'ondes analogiques pour donner plus de richesse et augmenter la versatilité du synthétiseur. Voici un graphique résumant le fonctionnement du module ainsi que la manière qu'il a de communiquer avec le synthétiseur.

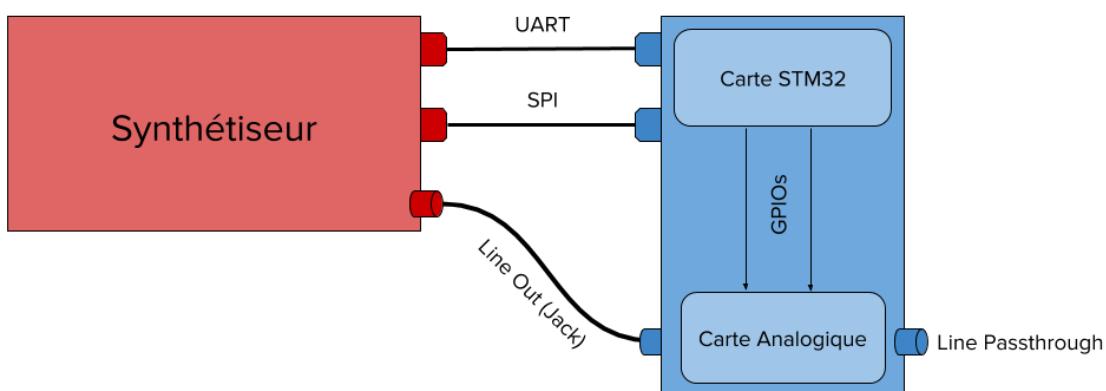


Figure 12.4: Fonctionnement du module et méthodes de communication utilisées

- La liaison UART permet à la carte STM32 de récupérer un numéro de note, ce numéro de note et ensuite converti en fréquence et permet de déterminer la fréquence des formes d'ondes numériques à générer par la STM32. Cette liaison permet aussi de récupérer la valeur du retard à créer par la carte analogique.
- La liaison SPI permet à la carte STM32 d'envoyer la forme d'onde numérique sur un des CNA du synthétiseur qui envoie ensuite cette dernière sur un multiplexeur

permettant de mixer à souhait forme d'onde numérique et analogique.

- La liaison line représente le signal path.

12.1.1 Ergonomie et utilisation

Nous avons donc décidé pour notre implémentation de l'interface HM de nous limiter au minimum de réglages possible par soucis de simplicité. Voici les réglages ainsi retenus :

- La durée de la réverbération : Length
- Le taux de réinjection : Depth
- L'égalisation permettant de filtrer les aigus : Tone

Voici donc la vision que nous avions de l'interface sur la carte Raspberry :

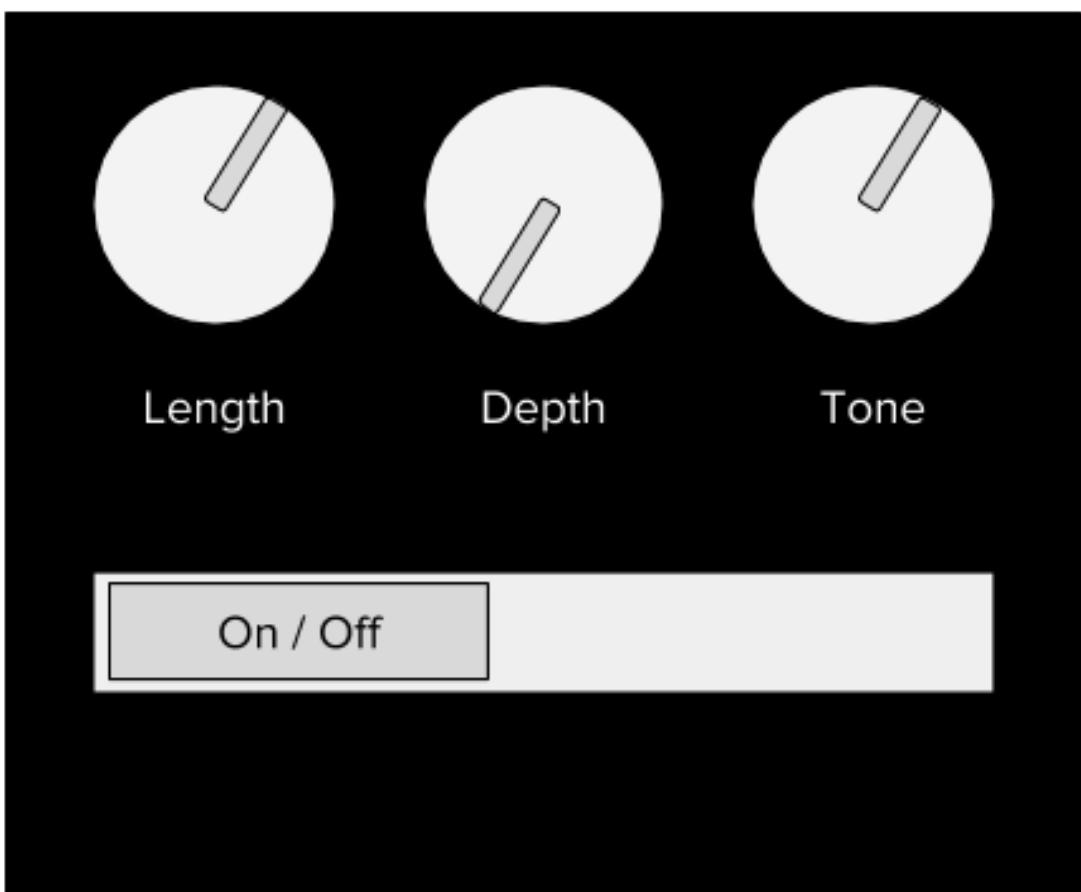


Figure 12.5: Prototype de l'interface graphique tel qu'imaginé au début du projet, non réalisé

Simple, épurée, claire, cette interface ne laisse pas de place à la confusion et permet donc de laisser place à sa créativité. Les boutons rotatifs tactiles peuvent être contrôlés par le toucher ou via un des codeurs physiques disposés sur la face avant du synthétiseur, en dessous de l'écran tactile, ou via l'écran tactile. C'est aussi depuis cette interface que le musicien peut préparer ses presets avant une performance live. L'interface imaginée devait se faire en JavaFX mais ne sera finalement pas réalisée.

12.1.2 Cahier des charges

Nous pouvons dégager les contraintes suivantes pour notre cahier des charges :

- Les caractéristiques Depth, Length et Tone doivent pouvoir être commandés depuis l'interface graphique.
- Les caractéristiques Depth, Length et Tone doivent avoir une plage de variation utile pour une application musicale :
 - Length : de 0 à 3s (delay de 500ms à 1s et reverb à plus d'une seconde).
 - Depth : de 0 à 100% (Ex : 20% envoie des données audio retardées à un cinquième du volume d'origine, créant des échos à atténuation progressive).
 - Tone : Fréquence de coupure variant de 500 à 2500Hz (plage de fréquence des mediums et aigus dans un synthétiseur)
- La communication entre la carte STM32 et le synthétiseur doit se faire via le bus SPI et la liaison UART.
- Il doit être possible d'enregistrer des presets pour les valeurs de Depth et Length.

12.2 Fonctionnement du circuit

12.2.1 Schéma global et fonctionnement

Au coeur de notre module de reverb/delay, se trouve le MN3009, composant intégré analogique, une BBD. Expliquons le fonctionnement de cette ligne à retard.

La BBD (Bridge Bucket Delay) est constituée d'une rangée de condensateurs mis les uns à la suite des autres. Le principe est simple: un premier condensateur reçoit le signal, se charge avec, puis le décharge dans un second condensateur et ainsi de suite comme un seau d'eau que l'on ferait passer à travers une ligne de personnes; les interrupteurs entre les différents condensateurs s'ouvrent et se ferment au rythme de deux horloges déphasées. Ainsi, plus il y a d'étages de condensateurs, plus le signal pourra être retardé.

256 pour les 256 étages de condensateurs de notre modèle MN3009. Nous pouvons ainsi résumer les caractéristiques de notre composant:

- Étages de retard: 256
- Fréquence de l'horloge: 10Khz à 200Khz
- Retard réalisé : 0.64ms à 12.8ms

Afin d'obtenir un son exploitable, le MN3009 a besoin d'être associé à un certain nombres de filtres dont nous allons détailler l'utilité. Voici, le schéma global dans lequel s'insère notre ligne à retard:

12.2.2 Filtre amont - Passe bas Anti-aliasing

Il s'agit d'un filtre passe-bas de fréquence de coupure 2.5 kHz utilisé pour l'anti-aliasing. Les deux montages à AOOp cascadés nous permettent d'obtenir un ordre 5. D'abord réalisé et testé sur breadboard ; nous l'utilisons finalement à la place sur filtre aval (rose) car son bon fonctionnement a été vérifié et il était primordial de nous débarrasser du bruit induit par les horloges de la BBD. Le problème de l'anti-aliasing en amont reste donc à être corrigé. Ce problème d'anti-aliasing vient du fait que la BBD travaille en temps discret. Le théorème de Shannon sur l'échantillonnage doit donc être respecté : $f_e > 2f_{max}$, en filtrant les fréquences trop hautes. Autrement, le phénomène de repliement de spectre apparaît. Avec une fréquence de coupure à 2.5 kHz, on s'assure ainsi de respecter le théorème, en effet au minimum: $f_e = 10 \text{ kHz}$ (fréquence de notre horloge BBD).

12.2.3 Filtre aval - De reconstruction du signal

C'est un filtre passe-bas d'ordre 4 également de fréquence de coupure d'environ 2.5kHz, utilisé pour la reconstruction du signal en effet, comme expliqué ci-dessus, le passage du signal dans la BDD génère un certain nombre de bruit: il nous faut encore une fois respecter le théorème de Shannon. Réalisé et testé sur breadboard, son fonctionnement n'était pas entièrement satisfaisant.

Nous avons à la place testé un MF6, filtre passe-bas du 6ème ordre de type Butterworth, que nous avons soudé sur notre carte de prototypage avec notre BBD. En comparant le son obtenu avec ce filtre et notre filtre amont, on constate que le filtre amont est plus efficace, c'est donc ce dernier que nous gardons en tant que filtre aval et qui est également soudé sur notre carte.

12.2.4 Réinjection

Le rebouclage ou réinjection est nécessaire pour réaliser les répétitions du son initial et ainsi imiter l'effet de réverbération naturelle du son qui se réfléchit sur des parois lisses et dures. Ce montage très simple constitué d'un AO et de quelques résistances est pourtant essentiel au bon fonctionnement de notre module d'effet.

12.2.5 Circuit de génération des horloges

Enfin, comme on a pu le constater précédemment, la BBD fonctionne avec des horloges. Un montage est prévu à cet effet dans la datasheet de notre composant mais l'objectif étant d'avoir une commande numérique, ce montage n'a donc pas été utilisé.

Nous utilisons plutôt des horloges générées via la carte STM32F411.

12.3 Réalisation

12.3.1 Schéma Eagle - S. Reynal

En plus des blocs décrits plus hauts, un composant ULN2803 a été ajouté en amont afin de faire passer les horloges de 3.3V aux 5V nécessaires à la BBD.

12.3.2 Programmation des horloges

Pour réaliser la commande de la BBD via une interface graphique, il est nécessaire de générer deux créneaux respectant le timing suivant :

Quand la fréquence d'horloge est grande, Length est court et vice versa. Afin de profiter un maximum de la plage de fonctionnement de notre BBD, il est nécessaire de générer deux créneaux d'une fréquence allant de 10kHz à 100kHz.

Nous avons d'abord expérimenté la génération de ces horloges en utilisant les Timers de la Raspberry Pi. Nous avons néanmoins abandonné cette implémentation afin de libérer le plus de ressources dans notre carte RaspBerry pour gérer l'interface graphique de manière fluide.

Ainsi la programmation de l'horloge du module a été transférée sur STM32.

Voici le code utilisé et commenté :

```
#include "mbed.h"
#include <stdio.h>
#include <stdlib.h>
```

```
#include <math.h>

DigitalOut CP1(PA_5); // GPIO pour Horloge 1
DigitalOut CP2(PA_6); // GPIO pour Horloge 2
InterruptIn button(PC_13); // Bouton Utilisateur
Ticker timer; // Ticker permettant d'appeler
// regulierement la fonction de generation des horloges

int flag=0; // drapeau init a 0 et passe a 1 a chaque changement de niveau

float nop_us = 0.5; // Leger delay en us,
// necessaire pour respecter le Timing de la Datasheet de la BBD

//----- Fonction generant les horloges

void clk(void){
    if(flag==0){
        CP1=!CP1;
        wait_us(nop_us);
        CP2=!CP2;
        flag=1;
    }
    else{
        CP2=!CP2;
        wait_us(nop_us);
        CP1=!CP1;
        flag=0;
    }
}

//----- Fonction qui attache un handler au Ticker

void classic_delay(int half_period_us){
    timer.attach_us(&clk, half_period_us);
}

//----- Main
int main(void){
    CP1=0;
    CP2=1;
    button.fall(&classic_delay(10000));
}
```

```
    return 0;  
}
```

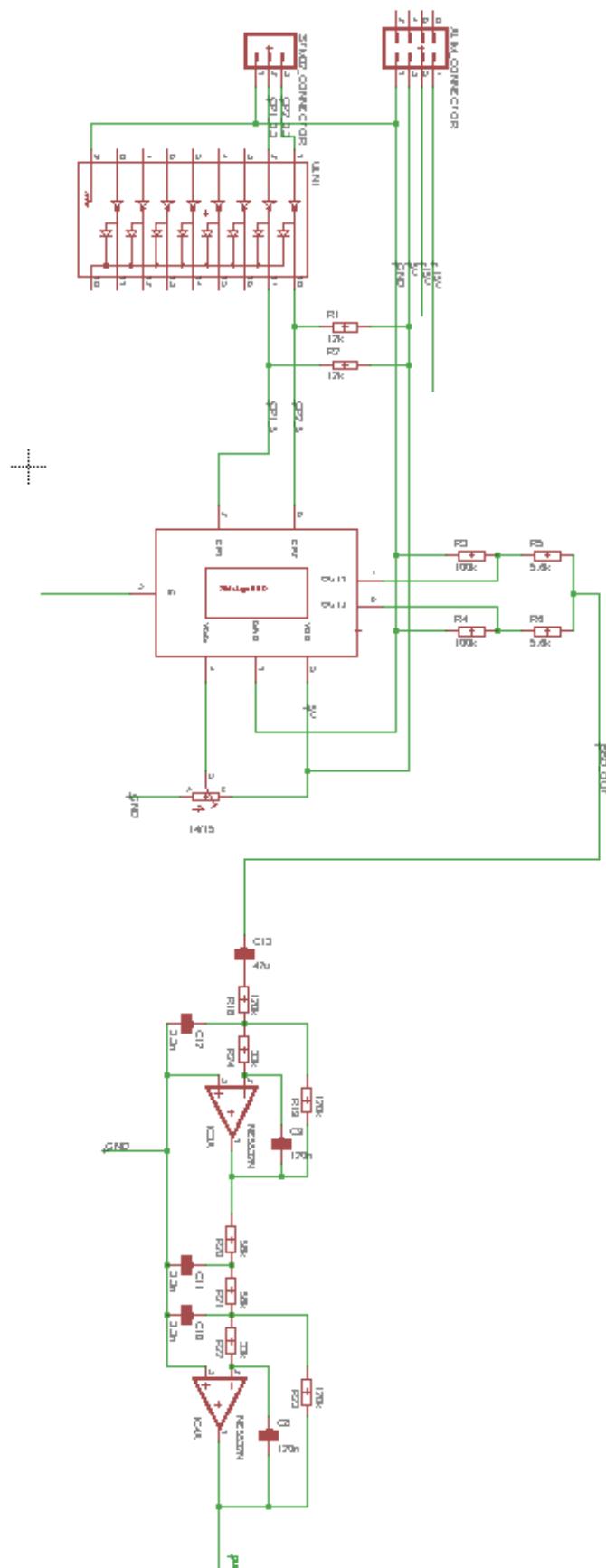
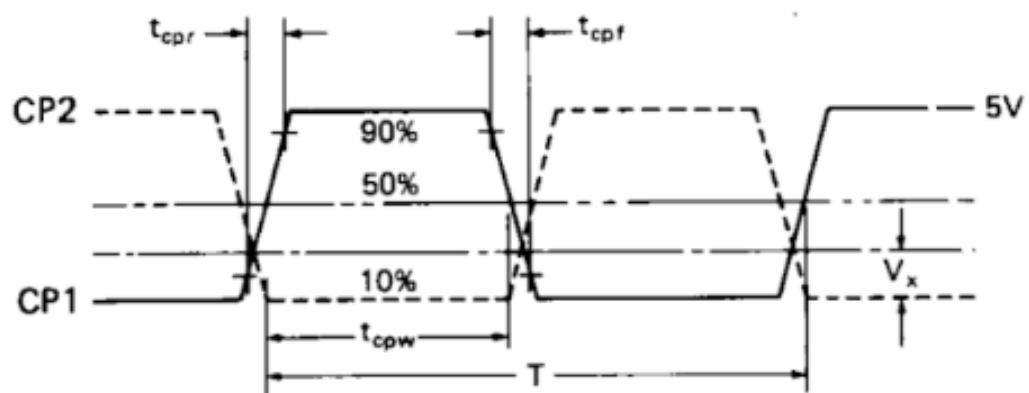


Figure 12.6: Schéma Eagle du montage de la BBD réalisé



Notre PULSAR comprend de nombreuses parties codées numériquement (Génération des enveloppes ADSR,wavetable, interface homme-machine etc..), et un des problèmes majeurs est la compatibilité entre ces codes.Plus précisément, la majorité des parties sont codées en C (la wavetable par exemple), et une partie en Java (JavaFX pour l'interface homme-machine (IHM)).Ainsi une question primordiale apparaît:

- **Comment rendre compatible du Java avec du C?**

La réponse est toute simple: c'est la **Java Native Interface (JNI)** qui va s'en occuper.
Dans cette partie, je vais donc vous présenter la JNI, ainsi que sa mise en place.

La JNI c'est quoi? Il s'agit d'une interface de programmation qui permet à du code Java (exécuté dans une JVM) de travailler, ou d'inter-changer des données, avec d'autres applications ou environnements **codés dans un autre langage, comme le C par exemple**: on parle de **code natif**.Ainsi l'IHM codé en JavaFX pourra être implémenter dans notre PULSAR en C grâce à la JNI.

- **Comment ça fonctionne?**

La JNI utilise ce qu'on appelle **les méthodes natives**.Ce sont ces types de méthodes qui, appelées depuis une machine virtuelle comme la JVM, permettront à du code développé en Java d'être traduit en C.

- **Pourquoi préférer le C au Java?**

Comme dit précédemment, la raison majeure est par soucis de compatibilité avec les autres modules codés en C. Outre cela, les méthodes natives présentent de nombreux atouts non négligeable à savoir:

- **Les codes natifs sont jusqu'à 20 fois plus rapide que Java**
- **La gestion de la mémoire est beaucoup plus performante**

Je vais maintenant vous présenter l'environnement de la JNI, ainsi que la façon dont il faut procéder pour utiliser ces méthodes natives à travers un exemple basique.

Avant de commencer, la première étape consiste à créer, dans la JVM, une classe comportant une méthode native. Nous allons prendre comme exemple pour cette partie ce code ci-dessous:

```
public class Main {
    public native int intMethod(int i);
    public static void main(String[] args) {
        System.loadLibrary("Main");
        System.out.println(new Main().intMethod(2));
    }
}
```

Figure A.1: Classe Java utilisant une méthode native

Ci-dessus, nous pouvons voir plusieurs instructions, dont notamment:

- La méthode native "intMethod", dont le type est **native**, qui spécifie à la JNI que l'on va utiliser du code natif
- Nous devons charger la bibliothèque qui contiendra notre méthode native. Cela se fait par l'appel à la méthode **loadLibrary**
- Enfin nous faisons appel à notre méthode native avec un simple **println**

Nous allons maintenant coder notre fonction native en langage C. Cela se traduirait, par la définition, dans un fichier **main.c**, de la fonction "intMethod". Pour ce faire, on compile d'abord notre classe java, à l'aide du compilateur intégré à la JVM, avec cette ligne de commande: **javac -h class_name.java**, où **class_name** correspond, ici, à "Main", et l'option "-h" permet de générer le fichier "Main.h" contenant le prototype de notre méthode native.

Notre fichier "Main.h" généré, nous allons pouvoir passer à l'étape de codage de notre fonction. Pour cela, il nous suffit maintenant de créer un fichier, nommé "Main.c" qui ressemble à cela: Nous pouvons voir différentes instructions et méthodes dont:

```
#include <jni.h>
#include "Main.h"

JNIEXPORT jint JNICALL Java_Main_intMethod(
    JNIEnv *env, jobject obj, jint i) {
    return i * i;
}
```

Figure A.2: Visualisation du fichier Main.c

- les "include" des fichiers contenant les prototypes/bibliothèque nécessaires pour notre fichier
- L'instruction "JNIEXPORT jint JNICALL" qui permet de définir le corps de notre fonction "Java_Main_intMethod"
- 2 pointeurs en tant qu'arguments de notre fonction qui sont présents **quelque soit la fonction que l'on veut coder**: env de type JNIEnv qui permet d'avoir accès à toute les fonction de JNI, et obj de type jobject qui est une référence à l'objet Java dans lequel la méthode native a été déclarer

- Notre fonction retourne simplement le carré de l'argument donné à notre fonction native : **return i*i**
- Ainsi en compilant maintenant ce fichier nous obtenons "4" ce qui correspond bien au résultat attendu.

Annexe B.1: Code création des tables d'onde (N. Meerun)

Voici le code C qui a permis de créer des formes d'ondes à partir d'un fichier WAV.

```
#include <stdio.h>
#include <stdlib.h>
static double Te = 25e-6; // 25us => fe=40kHz
static double freq_note =440;
static int digvco_buffer_size;

// --- var globales ---

int loop_start , loop_end;
void init_buffer_digvco(char* file_name);
int main(int argc , char *argv[]){
int i;
printf('DIG_VCO: argument_ligne_de_commande=%s\n' , argv[1]);
init_buffer_digvco(argv[1]);

=====
Initialisation index de debut et fin
=====
loop_start=90;
loop_end=201;
int a;

=====
Affichage du Wave Header puis de l'indexation
```

```
=====
for_(i=0;i<44;i++){
printf ("\t%02x", digvco_buffer[ i ]);
}
printf("_\n");
printf("_\n");
for(a=loop_start;a<loop_end+1;a++){
printf ("_\t%02x", digvco_buffer[ a ]);
// printf("%i ",a);
}

printf(_"\n");
printf("_\n");

=====
Creation_de_l'onde
=====
a=loop_start;
int charg [ digvco_buffer_size ];
for(i=0;i<300;i++){
charg [ i ]=digvco_buffer[ a++ ];
if (a>loop_end){
a=loop_start;
}

printf ("\t_\%02x" , charg [ i ]);

}
printf("_\n");
return 0;
}
=====

ouverture du fichier WAV
=====

//Recherche du fichier WAV passe en argument

void init_buffer_digvco(char* file_name){
FILE *fp ;
int i , n;
fp = fopen (file_name , "rb");
if (fp == NULL){
printf("DIG_VCO:_erreur_ouverture_fichier_\n");
exit(0);
}
else {
```

```

printf("DIG_VCO:_ouverture_fichier_ok\n");}

// recherche longueur du fichier:

fseek(fp, 0, SEEK_END);
digvco_buffer_size = ftell(fp);
rewind(fp);
printf("DIG_VCO:_le_fichier_fait_%d_bytes\n", digvco_buffer_size);

// lecture fichier et remplissage du buffer

digvco_buffer = malloc(digvco_buffer_size * sizeof(*digvco_buffer));
n = fread(digvco_buffer, 1, digvco_buffer_size, fp);
printf("DIG_VCO:_read_%d_bytes\n", n);
fclose(fp);
i=0;

=====
codage des differents chunks du WAVE Header
=====

// Chunck ID:

printf("DIG_VCO:_ChunkID:_\t%c%c%c%c\n", digvco_buffer[i++],
digvco_buffer[i++], digvco_buffer[i++], digvco_buffer[i++]);

// Chunck SIZE:

printf("DIG_VCO:_ChunckSize:_\t%d_bytes\n", ((int)digvco_buffer[i++]
((int)digvco_buffer[i++] << 8) + ((int)digvco_buffer[i++] << 16)
+((int)digvco_buffer[i++] <<24));

// Chunck FORMAT:

printf("DIG_VCO:_FORMAT:_\t%c%c%c%c\n", digvco_buffer[i++],
digvco_buffer[i++], digvco_buffer[i++], digvco_buffer[i++]);

// Chunck Subchunk1ID

printf("DIG_VCO:_SUBCHUNK1ID:_\t%c%c%c%c\n",
digvco_buffer[i++], digvco_buffer[i++], digvco_buffer[i++], digvco_buffer[i++]);

// Subchunk1Size

printf("DIG_VCO:_SubChunck1_Size:_\t%d_bytes\n", ((int)digvco_buffer[i++]
+((int)digvco_buffer[i++] << 8) +((int)digvco_buffer[i++] << 16) +
+((int)digvco_buffer[i++] << 24)));

```

```

((int) digvco_buffer[ i++ ] <<24));

// AudioFormat

printf( "DIG_VCO_:AUDIOFormat_:
\ t_%02x_%02x_\n" , digvco_buffer[ i++ ],digvco_buffer[ i++ ]);

// NumChannels

printf( "DIG_VCO_:NUMChannels_:_\ t_%02x_\n" , digvco_buffer[ i++ ],
digvco_buffer[ i++ ]);

// SampleRate

printf( "DIG_VCO_:SampleRate_:_\ t_%d_Hz_\n" ,((int)digvco_buffer[ i++ ]) +
((int)digvco_buffer[ i++ ] << 8) + ((int)digvco_buffer[ i++ ] << 16) +
((int)digvco_buffer[ i++ ] <<24));

// ByteRate

printf( "DIG_VCO_:ByteRate_:_\ t_%d_Hz_\n" ,((int)digvco_buffer[ i++ ]) +
((int)digvco_buffer[ i++ ] << 8) + ((int)digvco_buffer[ i++ ] << 16) +
((int)digvco_buffer[ i++ ] <<24));

// BlockAlign

printf( "DIG_VCO_:BlockAlign_:_\ t_%d_bytes_\n" , ((int)digvco_buffer[ i++ ]) +
((int)digvco_buffer[ i++ ] << 8));

// BytesPerSample

printf( "DIG_VCO_:BitsperSample_:_\ t_%d_bits / sample_\n" ,
((int)digvco_buffer[ i++ ]) +((int)digvco_ buffer[ i++ ] << 8));
// LSB + MSB << 8 vs LSB << 8 + MSB

// Subchunk2ID

printf( "DIG_VCO_:Subchunk2ID_:
\ t_%c%c%c%c_\n" , digvco_buffer[ i++ ],digvco_buffer[ i++ ],digvco_buffer[ i++ ],
digvco_buffer[ i++ ]);

// Subchunk2Size

printf( "DIG_VCO_:SubChunck2_Size_:_\ t_%d_bytes_\n" ,((int)digvco_buffer[ i++ ]) +
((int)digvco_buffer[ i++ ] << 8) + ((int)digvco_buffer[ i++ ] << 16) +

```

```
((int)digvco_buffer[i++] <<24));
```


- CNA : MC4802
- Potentiomètre programmable : MCP41100
- Expander de bus GPIO : MCP32017
- Délay analogique : MN3209
- Triode intégrée : nutube
- OTA : LM13700
- Quad VCA : SSM2164
- VCF : CEM3320

**MICROCHIP**

MCP4802/4812/4822

8/10/12-Bit Dual Voltage Output Digital-to-Analog Converter with Internal V_{REF} and SPI Interface

Features

- MCP4802: Dual 8-Bit Voltage Output DAC
- MCP4812: Dual 10-Bit Voltage Output DAC
- MCP4822: Dual 12-Bit Voltage Output DAC
- Rail-to-Rail Output
- SPI Interface with 20 MHz Clock Support
- Simultaneous Latching of the Dual DACs with LDAC pin
- Fast Settling Time of 4.5 µs
- Selectable Unity or 2x Gain Output
- 2.048V Internal Voltage Reference
- 50 ppm/°C V_{REF} Temperature Coefficient
- 2.7V to 5.5V Single-Supply Operation
- Extended Temperature Range: -40°C to +125°C

Applications

- Set Point or Offset Trimming
- Sensor Calibration
- Precision Selectable Voltage Reference
- Portable Instrumentation (Battery-Powered)
- Calibration of Optical Communication Devices

Related Products⁽¹⁾

P/N	DAC Resolution	No. of Channels	Voltage Reference (V _{REF})
MCP4801	8	1	Internal (2.048V)
MCP4811	10	1	
MCP4821	12	1	
MCP4802	8	2	
MCP4812	10	2	
MCP4822	12	2	
MCP4901	8	1	External
MCP4911	10	1	
MCP4921	12	1	
MCP4902	8	2	
MCP4912	10	2	
MCP4922	12	2	

Note 1: The products listed here have similar AC/DC performances.

Description

The MCP4802/4812/4822 devices are dual 8-bit, 10-bit and 12-bit buffered voltage output Digital-to-Analog Converters (DACs), respectively. The devices operate from a single 2.7V to 5.5V supply with SPI compatible Serial Peripheral Interface.

The devices have a high precision internal voltage reference (V_{REF} = 2.048V). The user can configure the full-scale range of the device to be 2.048V or 4.096V by setting the Gain Selection Option bit (gain of 1 or 2).

Each DAC channel can be operated in Active or Shutdown mode individually by setting the Configuration register bits. In Shutdown mode, most of the internal circuits in the shutdown channel are turned off for power savings and the output amplifier is configured to present a known high resistance output load (500 kΩ, typical).

The devices include double-buffered registers, allowing synchronous updates of two DAC outputs using the LDAC pin. These devices also incorporate a Power-on Reset (POR) circuit to ensure reliable power-up.

The devices utilize a resistive string architecture, with its inherent advantages of low DNL error, low ratio metric temperature coefficient and fast settling time. These devices are specified over the extended temperature range (+125°C).

The devices provide high accuracy and low noise performance for consumer and industrial applications where calibration or compensation of signals (such as temperature, pressure and humidity) are required.

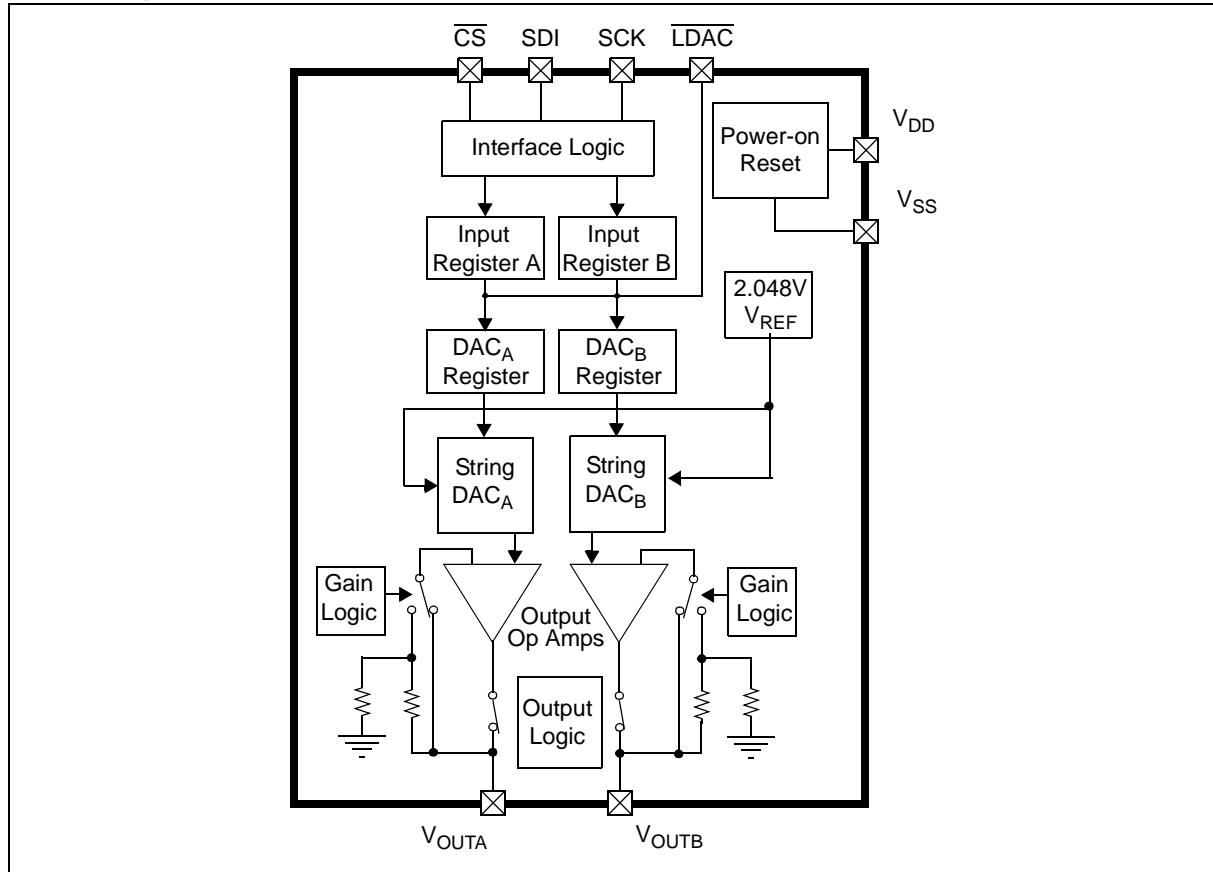
The MCP4802/4812/4822 devices are available in the PDIP, SOIC and MSOP packages.

Package Types

8-Pin PDIP, SOIC, MSOP							
V _{DD} [1]	●	8	V _{OUTA}				
CS [2]		7	V _{SS}				
SCK [3]		6	V _{OUTB}				
SDI [4]		5	LDAC				
MCP4802: 8-bit dual DAC MCP4812: 10-bit dual DAC MCP4822: 12-bit dual DAC							

MCP4802/4812/4822

Block Diagram



1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

V_{DD}	6.5V
All inputs and outputs	$V_{SS} - 0.3V$ to $V_{DD} + 0.3V$
Current at Input Pins	± 2 mA
Current at Supply Pins	± 50 mA
Current at Output Pins	± 25 mA
Storage temperature	-65°C to +150°C
Ambient temp. with power applied	-55°C to +125°C
ESD protection on all pins	≥ 4 kV (HBM), ≥ 400 V (MM)	
Maximum Junction Temperature (T_J)	+150°C

† **Notice:** Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 5V$, $V_{SS} = 0V$, $V_{REF} = 2.048V$, Output Buffer Gain (G) = 2x, $R_L = 5\text{ k}\Omega$ to GND, $C_L = 100\text{ pF}$, $T_A = -40$ to $+85^\circ\text{C}$. Typical values are at $+25^\circ\text{C}$.						
Parameters	Sym	Min	Typ	Max	Units	Conditions
Power Requirements						
Input Voltage	V_{DD}	2.7	—	5.5	V	
Input Current	I_{DD}	—	415	750	μA	All digital inputs are grounded, all analog outputs (V_{OUT}) are unloaded. Code = 0x000h
Software Shutdown Current	I_{SHDN_SW}	—	3.3	6	μA	
Power-on Reset Threshold	V_{POR}	—	2.0	—	V	
DC Accuracy						
MCP4802						
Resolution	n	8	—	—	Bits	
INL Error	INL	-1	± 0.125	1	LSb	
DNL	DNL	-0.5	± 0.1	+0.5	LSb	Note 1
MCP4812						
Resolution	n	10	—	—	Bits	
INL Error	INL	-3.5	± 0.5	3.5	LSb	
DNL	DNL	-0.5	± 0.1	+0.5	LSb	Note 1
MCP4822						
Resolution	n	12	—	—	Bits	
INL Error	INL	-12	± 2	12	LSb	
DNL	DNL	-0.75	± 0.2	+0.75	LSb	Note 1
Offset Error	V_{OS}	-1	± 0.02	1	% of FSR	Code = 0x000h
Offset Error Temperature Coefficient	$V_{OS}/^\circ\text{C}$	—	0.16	—	ppm/ $^\circ\text{C}$	-45°C to +25°C
		—	-0.44	—	ppm/ $^\circ\text{C}$	+25°C to +85°C
Gain Error	g_E	-2	-0.10	2	% of FSR	Code = 0xFFFFh, not including offset error
Gain Error Temperature Coefficient	$\Delta g/\text{ }^\circ\text{C}$	—	-3	—	ppm/ $^\circ\text{C}$	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

MCP4802/4812/4822

ELECTRICAL CHARACTERISTICS (CONTINUED)

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 5V$, $V_{SS} = 0V$, $V_{REF} = 2.048V$, Output Buffer Gain (G) = 2x, $R_L = 5\text{ k}\Omega$ to GND, $C_L = 100\text{ pF}$, $T_A = -40$ to $+85^\circ\text{C}$. Typical values are at $+25^\circ\text{C}$.						
Parameters	Sym	Min	Typ	Max	Units	Conditions
Internal Voltage Reference (V_{REF})						
Internal Reference Voltage	V_{REF}	2.008	2.048	2.088	V	V_{OUTA} when $G = 1x$ and Code = 0xFFFF
Temperature Coefficient (Note 2)	$\Delta V_{REF}/^\circ\text{C}$	—	125	325	ppm/ $^\circ\text{C}$	-40 $^\circ\text{C}$ to 0 $^\circ\text{C}$
		—	0.25	0.65	LSb/ $^\circ\text{C}$	-40 $^\circ\text{C}$ to 0 $^\circ\text{C}$
		—	45	160	ppm/ $^\circ\text{C}$	0 $^\circ\text{C}$ to +85 $^\circ\text{C}$
		—	0.09	0.32	LSb/ $^\circ\text{C}$	0 $^\circ\text{C}$ to +85 $^\circ\text{C}$
Output Noise (V_{REF} Noise)	E_{NREF} (0.1-10 Hz)	—	290	—	$\mu\text{V}_{\text{p-p}}$	Code = 0xFFFF, $G = 1x$
Output Noise Density	e_{NREF} (1 kHz)	—	1.2	—	$\mu\text{V}/\sqrt{\text{Hz}}$	Code = 0xFFFF, $G = 1x$
	e_{NREF} (10 kHz)	—	1.0	—	$\mu\text{V}/\sqrt{\text{Hz}}$	Code = 0xFFFF, $G = 1x$
1/f Corner Frequency	f_{CORNER}	—	400	—	Hz	
Output Amplifier						
Output Swing	V_{OUT}	—	0.01 to $V_{DD}-0.04$	—	V	Accuracy is better than 1 LSb for $V_{OUT} = 10\text{ mV}$ to $(V_{DD}-40\text{ mV})$
Phase Margin	PM	—	66	—	Degree ($^\circ$)	$C_L = 400\text{ pF}$, $R_L = \infty$
Slew Rate	SR	—	0.55	—	V/ μs	
Short Circuit Current	I_{SC}	—	15	24	mA	
Settling Time	t_{SETTLING}	—	4.5	—	μs	Within 1/2 LSb of final value from 1/4 to 3/4 full-scale range
Dynamic Performance (Note 2)						
DAC-to-DAC Crosstalk		—	<10	—	nV-s	
Major Code Transition Glitch		—	45	—	nV-s	1 LSb change around major carry (0111...1111 to 1000...0000)
Digital Feedthrough		—	<10	—	nV-s	
Analog Crosstalk		—	<10	—	nV-s	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

MCP4802/4812/4822

ELECTRICAL CHARACTERISTIC WITH EXTENDED TEMPERATURE

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 5V$, $V_{SS} = 0V$, $V_{REF} = 2.048V$, Output Buffer Gain (G) = $2x$, $R_L = 5\text{ k}\Omega$ to GND, $C_L = 100\text{ pF}$. Typical values are at $+125^\circ\text{C}$ by characterization or simulation.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Power Requirements						
Input Voltage	V_{DD}	2.7	—	5.5	V	
Input Current	I_{DD}	—	440	—	μA	All digital inputs are grounded, all analog outputs (V_{OUT}) are unloaded. Code = 0x000h.
Software Shutdown Current	I_{SHDN_SW}	—	5	—	μA	
Power-On Reset threshold	V_{POR}	—	1.85	—	V	
DC Accuracy						
MCP4802						
Resolution	n	8	—	—	Bits	
INL Error	INL	—	± 0.25	—	LSb	
DNL	DNL	—	± 0.2	—	LSb	Note 1
MCP4812						
Resolution	n	10	—	—	Bits	
INL Error	INL	—	± 1	—	LSb	
DNL	DNL	—	± 0.2	—	LSb	Note 1
MCP4822						
Resolution	n	12	—	—	Bits	
INL Error	INL	—	± 4	—	LSb	
DNL	DNL	—	± 0.25	—	LSb	Note 1
Offset Error	V_{OS}	—	± 0.02	—	% of FSR	Code = 0x000h
Offset Error Temperature Coefficient	$V_{OS}/^\circ\text{C}$	—	-5	—	ppm/ $^\circ\text{C}$	+25 $^\circ\text{C}$ to +125 $^\circ\text{C}$
Gain Error	g_E	—	-0.10	—	% of FSR	Code = 0xFFFFh, not including offset error
Gain Error Temperature Coefficient	$\Delta g/\text{ }^\circ\text{C}$	—	-3	—	ppm/ $^\circ\text{C}$	
Internal Voltage Reference (V_{REF})						
Internal Reference Voltage	V_{REF}	—	2.048	—	V	V_{OUTA} when $G = 1x$ and Code = 0xFFFFh
Temperature Coefficient (Note 2)	$\Delta V_{REF}/^\circ\text{C}$	—	125	—	ppm/ $^\circ\text{C}$	-40 $^\circ\text{C}$ to 0 $^\circ\text{C}$
		—	0.25	—	LSb/ $^\circ\text{C}$	-40 $^\circ\text{C}$ to 0 $^\circ\text{C}$
		—	45	—	ppm/ $^\circ\text{C}$	0 $^\circ\text{C}$ to +85 $^\circ\text{C}$
		—	0.09	—	LSb/ $^\circ\text{C}$	0 $^\circ\text{C}$ to +85 $^\circ\text{C}$
Output Noise (V_{REF} Noise)	E_{NREF} (0.1 – 10 Hz)	—	290	—	$\mu\text{V}_{\text{p-p}}$	Code = 0xFFFFh, $G = 1x$
Output Noise Density	e_{NREF} (1 kHz)	—	1.2	—	$\mu\text{V}/\sqrt{\text{Hz}}$	Code = 0xFFFFh, $G = 1x$
	e_{NREF} (10 kHz)	—	1.0	—	$\mu\text{V}/\sqrt{\text{Hz}}$	Code = 0xFFFFh, $G = 1x$
1/f Corner Frequency	f_{CORNER}	—	400	—	Hz	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

MCP4802/4812/4822

ELECTRICAL CHARACTERISTIC WITH EXTENDED TEMPERATURE (CONTINUED)

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 5V$, $V_{SS} = 0V$, $V_{REF} = 2.048V$, Output Buffer Gain (G) = 2x, $R_L = 5 k\Omega$ to GND, $C_L = 100 pF$. Typical values are at $+125^\circ C$ by characterization or simulation.						
Output Amplifier						
Parameters	Sym	Min	Typ	Max	Units	Conditions
Output Amplifier						
Output Swing	V_{OUT}	—	0.01 to $V_{DD} - 0.04$	—	V	Accuracy is better than 1 LSb for $V_{OUT} = 10 \text{ mV}$ to $(V_{DD} - 40 \text{ mV})$
Phase Margin	PM	—	66	—	Degree (°)	$C_L = 400 \text{ pF}$, $R_L = \infty$
Slew Rate	SR	—	0.55	—	V/ μ s	
Short Circuit Current	I_{SC}	—	17	—	mA	
Settling Time	$t_{SETTLING}$	—	4.5	—	μ s	Within 1/2 LSb of final value from 1/4 to 3/4 full-scale range
Dynamic Performance (Note 2)						
DAC-to-DAC Crosstalk		—	<10	—	nV-s	
Major Code Transition Glitch		—	45	—	nV-s	1 LSb change around major carry (0111...1111 to 1000...0000)
Digital Feedthrough		—	<10	—	nV-s	
Analog Crosstalk		—	<10	—	nV-s	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

AC CHARACTERISTICS (SPI TIMING SPECIFICATIONS)

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 2.7V - 5.5V$, $T_A = -40$ to $+125^\circ C$. Typical values are at $+25^\circ C$.						
Parameters	Sym	Min	Typ	Max	Units	Conditions
Schmitt Trigger High-Level Input Voltage (All digital input pins)	V_{IH}	0.7 V_{DD}	—	—	V	
Schmitt Trigger Low-Level Input Voltage (All digital input pins)	V_{IL}	—	—	0.2 V_{DD}	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	0.05 V_{DD}	—	V	
Input Leakage Current	$I_{LEAKAGE}$	-1	—	1	μ A	$\overline{LDAC} = \overline{CS} = \overline{SDI} = \overline{SCK} = V_{DD}$ or V_{SS}
Digital Pin Capacitance (All inputs/outputs)	C_{IN} , C_{OUT}	—	10	—	pF	$V_{DD} = 5.0V$, $T_A = +25^\circ C$, $f_{CLK} = 1 \text{ MHz}$ (Note 1)
Clock Frequency	F_{CLK}	—	—	20	MHz	$T_A = +25^\circ C$ (Note 1)
Clock High Time	t_{HI}	15	—	—	ns	Note 1
Clock Low Time	t_{LO}	15	—	—	ns	Note 1
\overline{CS} Fall to First Rising CLK Edge	t_{CSSR}	40	—	—	ns	Applies only when \overline{CS} falls with CLK high. (Note 1)
Data Input Setup Time	t_{SU}	15	—	—	ns	Note 1
Data Input Hold Time	t_{HD}	10	—	—	ns	Note 1
SCK Rise to \overline{CS} Rise Hold Time	t_{CHS}	15	—	—	ns	Note 1

Note 1: This parameter is ensured by design and not 100% tested.

MCP4802/4812/4822

AC CHARACTERISTICS (SPI TIMING SPECIFICATIONS)

Electrical Specifications: Unless otherwise indicated, $V_{DD} = 2.7V - 5.5V$, $T_A = -40$ to $+125^{\circ}\text{C}$.
Typical values are at $+25^{\circ}\text{C}$.

Parameters	Sym	Min	Typ	Max	Units	Conditions
CS High Time	t_{CSH}	15	—	—	ns	Note 1
LDAC Pulse Width	t_{LD}	100	—	—	ns	Note 1
LDAC Setup Time	t_{LS}	40	—	—	ns	Note 1
SCK Idle Time before CS Fall	t_{IDLE}	40	—	—	ns	Note 1

Note 1: This parameter is ensured by design and not 100% tested.

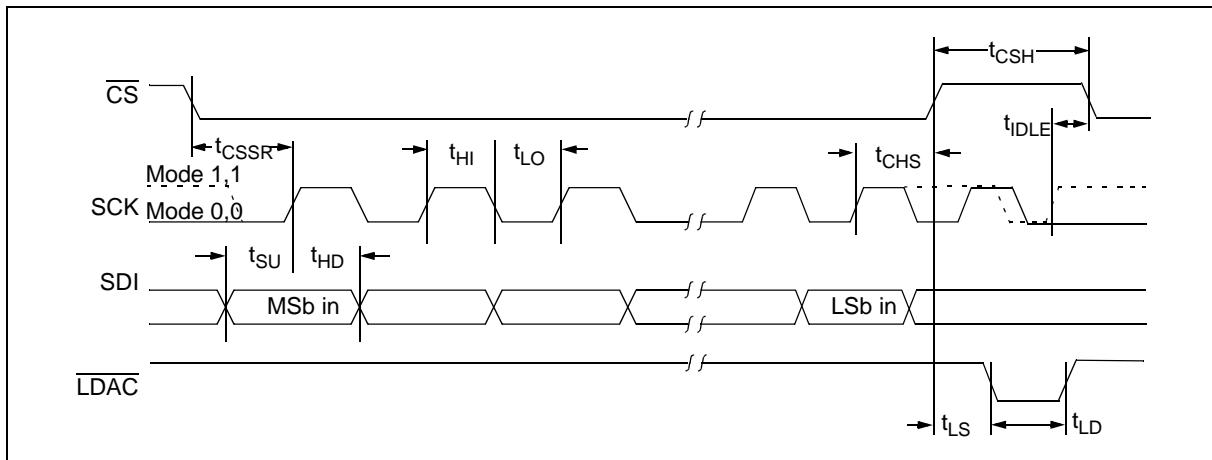


FIGURE 1-1: SPI Input Timing Data.

TEMPERATURE CHARACTERISTICS

Electrical Specifications: Unless otherwise indicated, $V_{DD} = +2.7\text{V}$ to $+5.5\text{V}$, $V_{SS} = \text{GND}$.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Temperature Ranges						
Specified Temperature Range	T_A	-40	—	+125	°C	
Operating Temperature Range	T_A	-40	—	+125	°C	Note 1
Storage Temperature Range	T_A	-65	—	+150	°C	
Thermal Package Resistances						
Thermal Resistance, 8L-MSOP	θ_{JA}	—	211	—	°C/W	
Thermal Resistance, 8L-PDIP	θ_{JA}	—	90	—	°C/W	
Thermal Resistance, 8L-SOIC	θ_{JA}	—	150	—	°C/W	

Note 1: The MCP4802/4812/4822 devices operate over this extended temperature range, but with reduced performance. Operation in this range must not cause T_J to exceed the maximum junction temperature of $+150^{\circ}\text{C}$.

MCP4802/4812/4822

NOTES:

2.0 TYPICAL PERFORMANCE CURVES

Note: The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{ k}\Omega$, $C_L = 100\text{ pF}$.

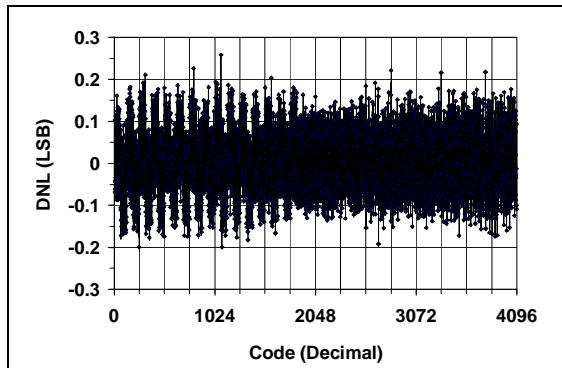


FIGURE 2-1: DNL vs. Code (MCP4822).

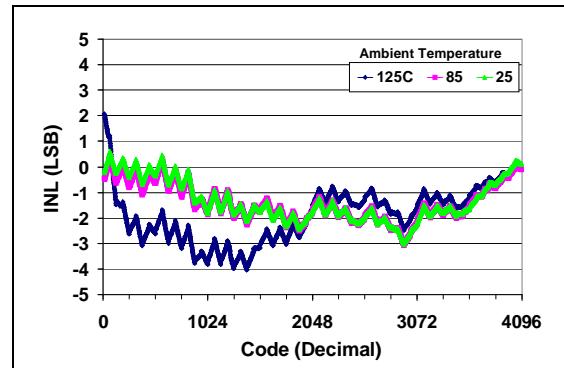


FIGURE 2-4: INL vs. Code and Temperature (MCP4822).

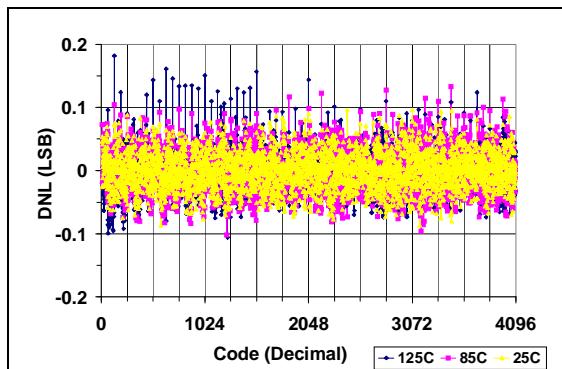


FIGURE 2-2: DNL vs. Code and Temperature (MCP4822).

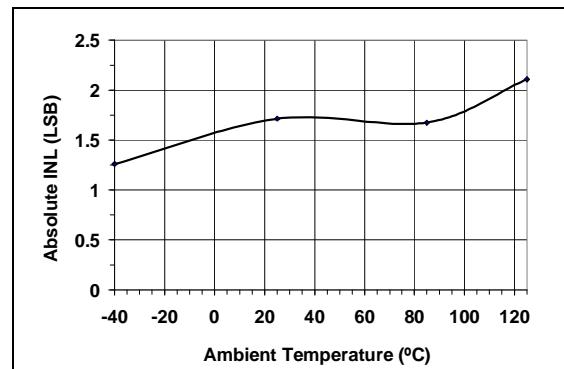


FIGURE 2-5: Absolute INL vs. Temperature (MCP4822).

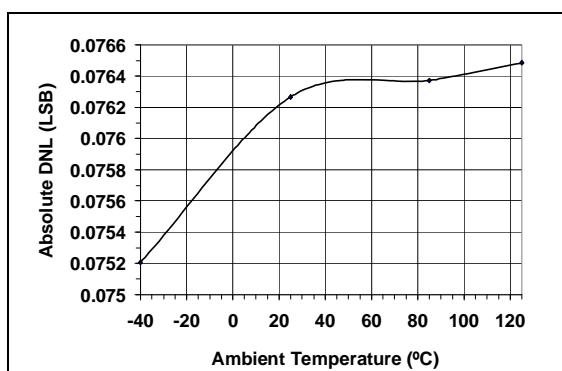


FIGURE 2-3: Absolute DNL vs. Temperature (MCP4822).

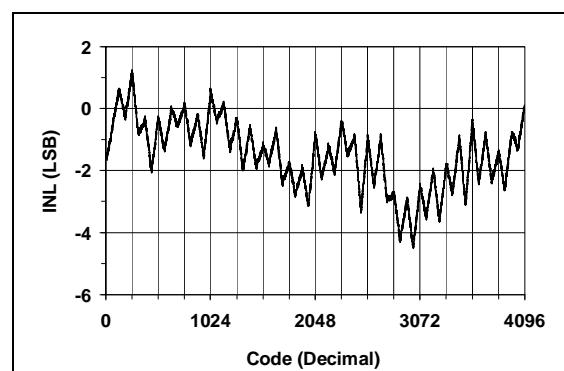


FIGURE 2-6: INL vs. Code (MCP4822).

Note: Single device graph for illustration of 64 code effect.

MCP4802/4812/4822

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{k}\Omega$, $C_L = 100\text{ pF}$.

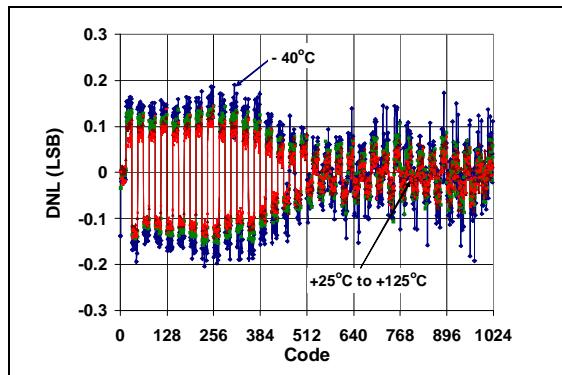


FIGURE 2-7: DNL vs. Code and Temperature (MCP4812).

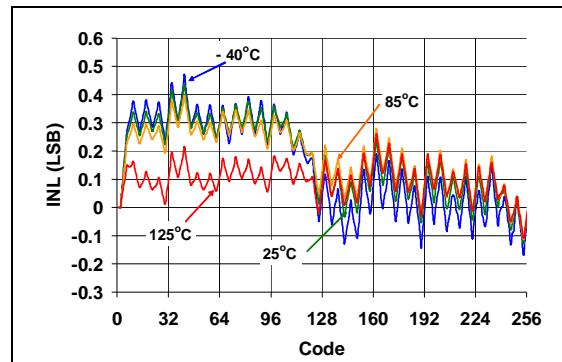


FIGURE 2-10: INL vs. Code and Temperature (MCP4802).

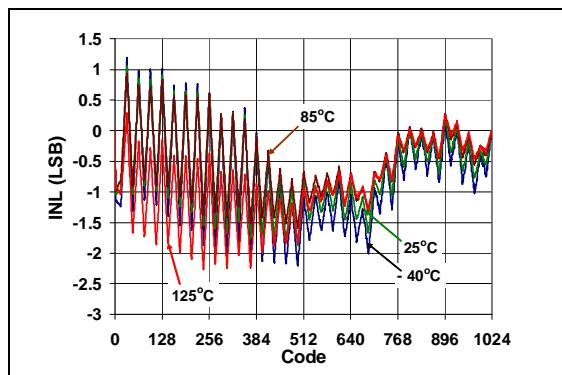


FIGURE 2-8: INL vs. Code and Temperature (MCP4812).

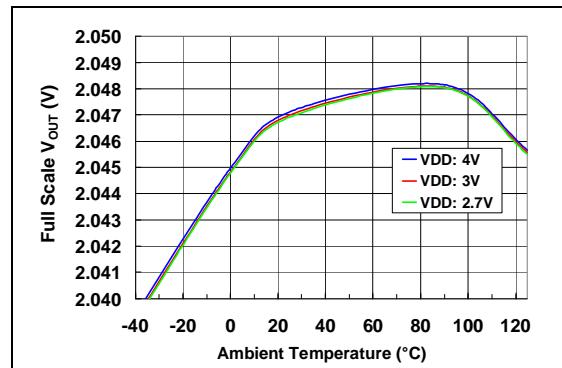


FIGURE 2-11: Full-Scale V_{OUTA} vs. Ambient Temperature and V_{DD} . Gain = 1x.

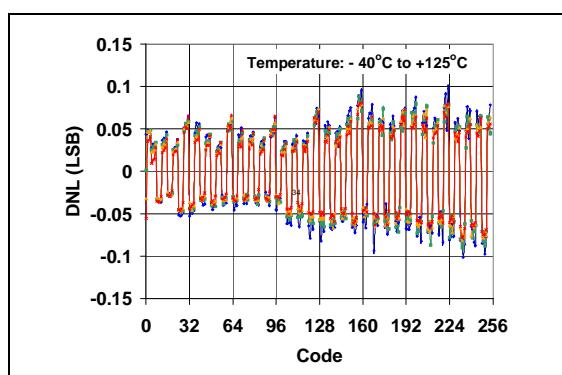


FIGURE 2-9: DNL vs. Code and Temperature (MCP4802).

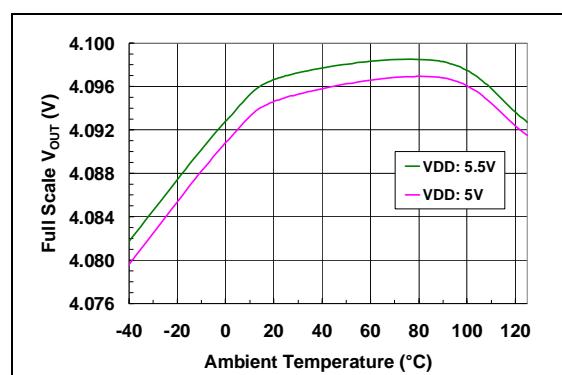


FIGURE 2-12: Full-Scale V_{OUTA} vs. Ambient Temperature and V_{DD} . Gain = 2x.

MCP4802/4812/4822

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{ k}\Omega$, $C_L = 100\text{ pF}$.

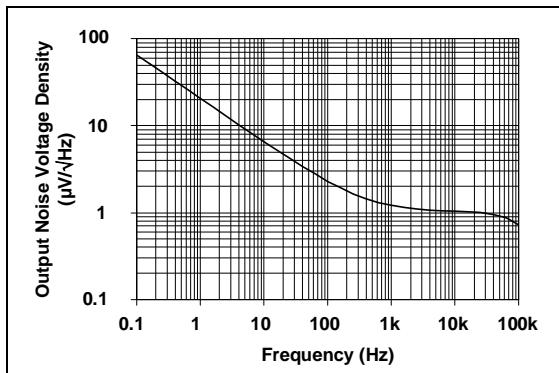


FIGURE 2-13: Output Noise Voltage Density (V_{REF} Noise Density) vs. Frequency. Gain = 1x.

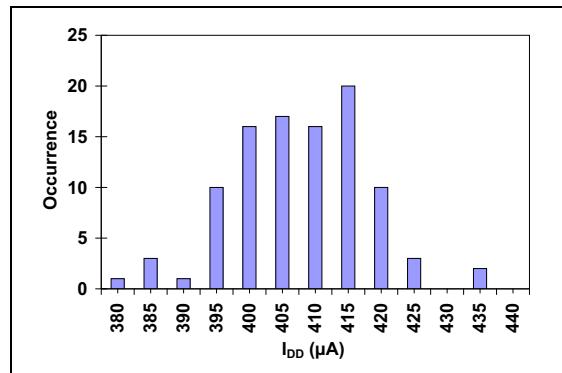


FIGURE 2-16: I_{DD} Histogram ($V_{DD} = 2.7\text{V}$).

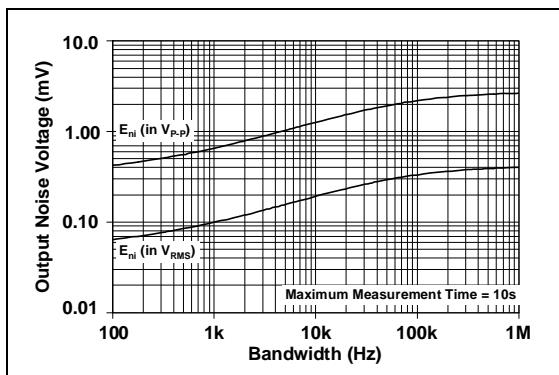


FIGURE 2-14: Output Noise Voltage (V_{REF} Noise Voltage) vs. Bandwidth. Gain = 1x.

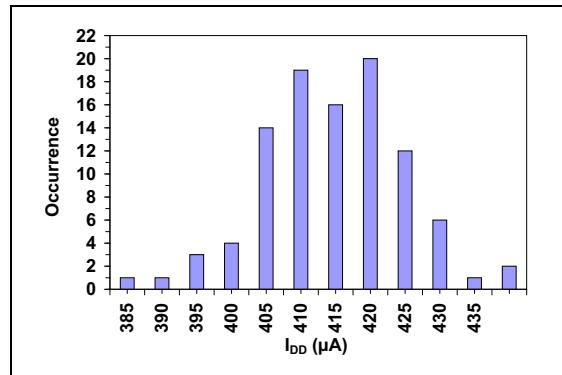


FIGURE 2-17: I_{DD} Histogram ($V_{DD} = 5.0\text{V}$).

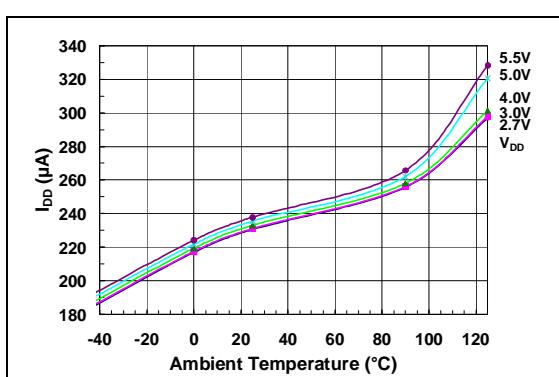


FIGURE 2-15: I_{DD} vs. Temperature and V_{DD} .

MCP4802/4812/4822

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{k}\Omega$, $C_L = 100\text{ pF}$.

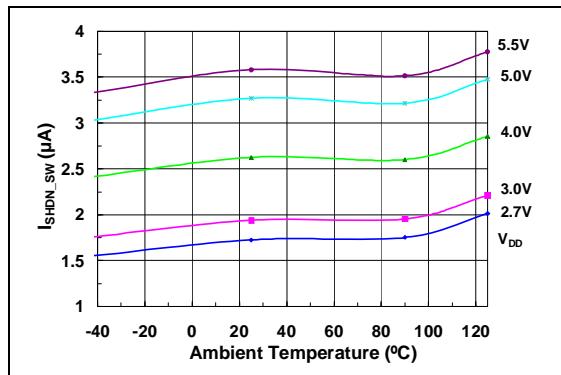


FIGURE 2-18: Software Shutdown Current vs. Temperature and V_{DD} .

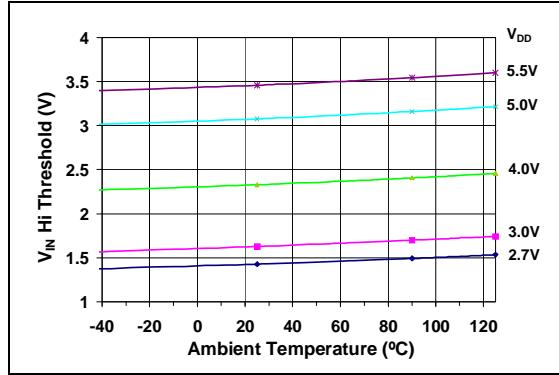


FIGURE 2-21: V_{IN} High Threshold vs. Temperature and V_{DD} .

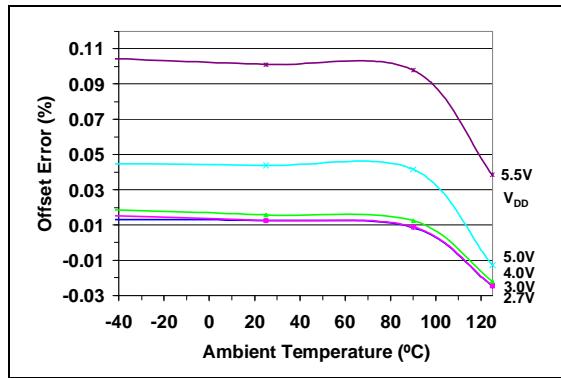


FIGURE 2-19: Offset Error vs. Temperature and V_{DD} .

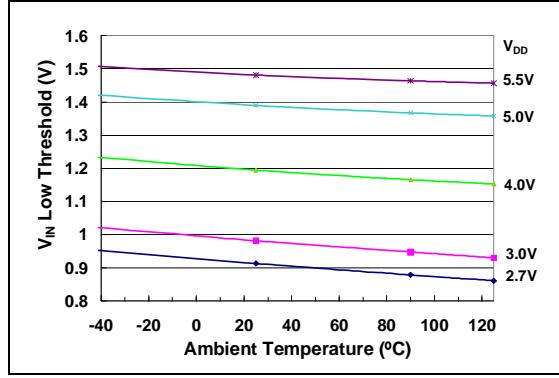


FIGURE 2-22: V_{IN} Low Threshold vs. Temperature and V_{DD} .

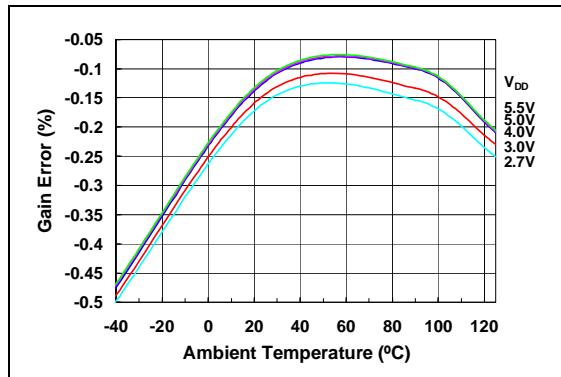


FIGURE 2-20: Gain Error vs. Temperature and V_{DD} .

MCP4802/4812/4822

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{ k}\Omega$, $C_L = 100\text{ pF}$.

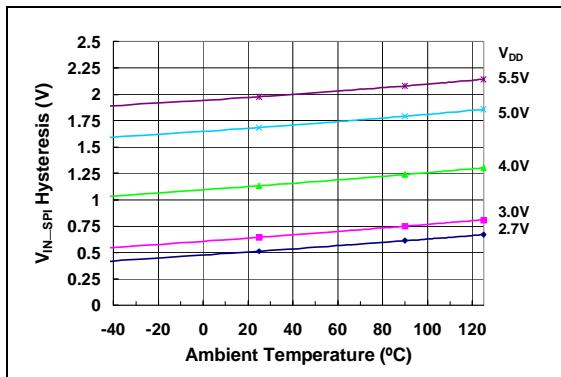


FIGURE 2-23: V_{IN_SPI} Hysteresis vs. Temperature and V_{DD} .

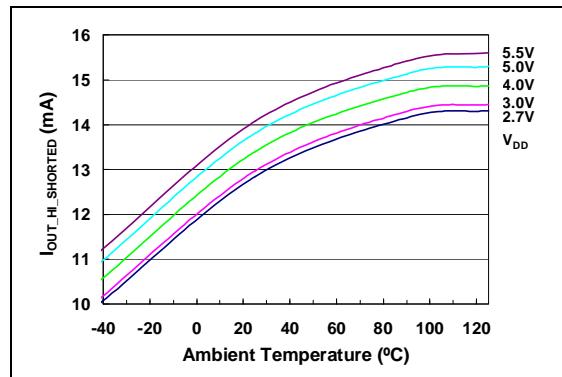


FIGURE 2-26: I_{OUT} High Short vs. Temperature and V_{DD} .

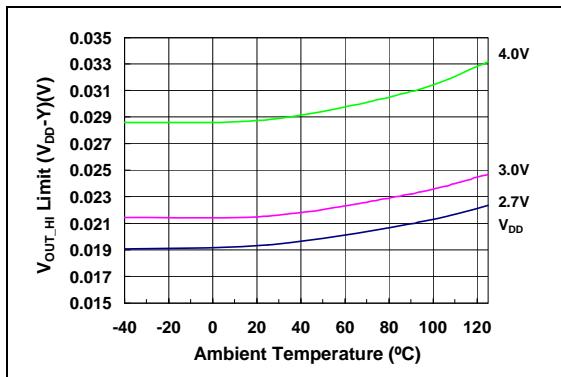


FIGURE 2-24: V_{OUT} High Limit vs. Temperature and V_{DD} .

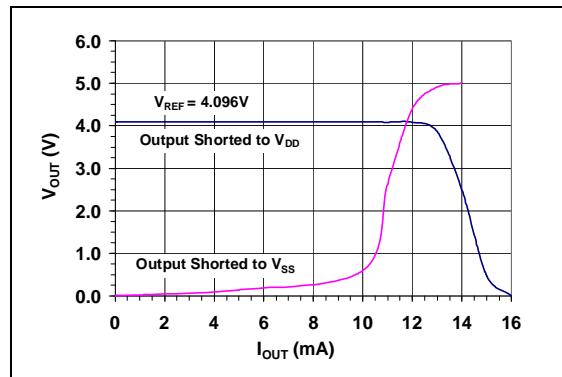


FIGURE 2-27: I_{OUT} vs. V_{OUT} . Gain = 2x.

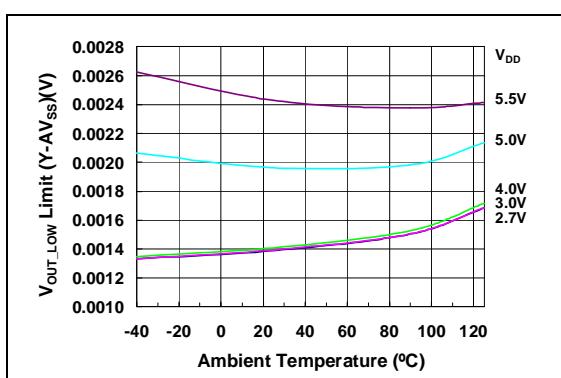


FIGURE 2-25: V_{OUT} Low Limit vs. Temperature and V_{DD} .

MCP4802/4812/4822

Note: Unless otherwise indicated, $T_A = +25^\circ\text{C}$, $V_{DD} = 5\text{V}$, $V_{SS} = 0\text{V}$, $V_{REF} = 2.048\text{V}$, Gain = 2x, $R_L = 5\text{k}\Omega$, $C_L = 100\text{ pF}$.

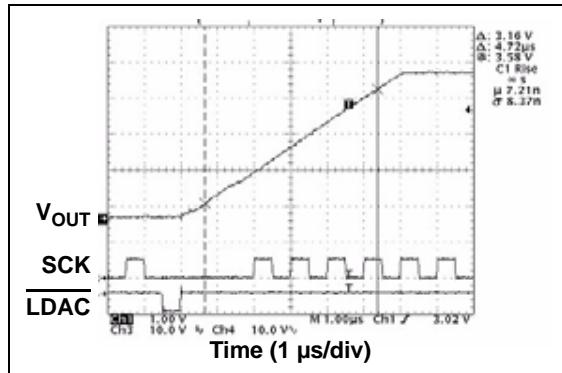


FIGURE 2-28: V_{OUT} Rise Time.

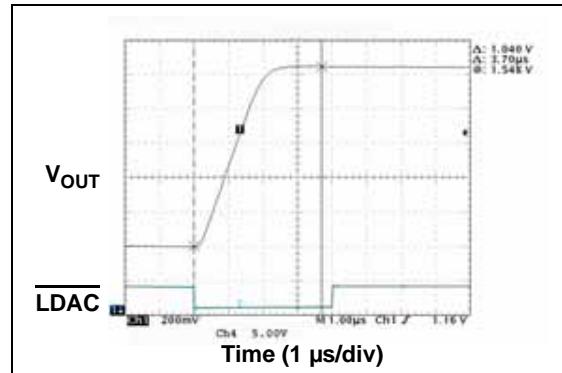


FIGURE 2-31: V_{OUT} Rise Time.

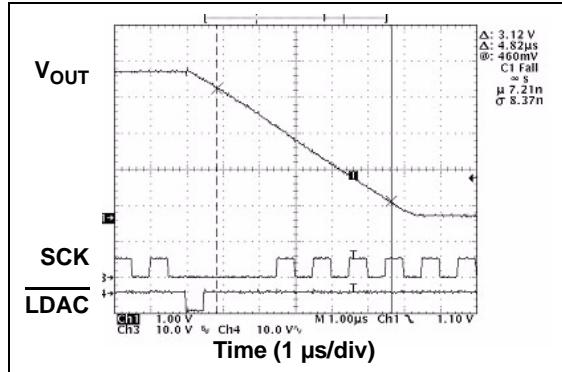


FIGURE 2-29: V_{OUT} Fall Time.

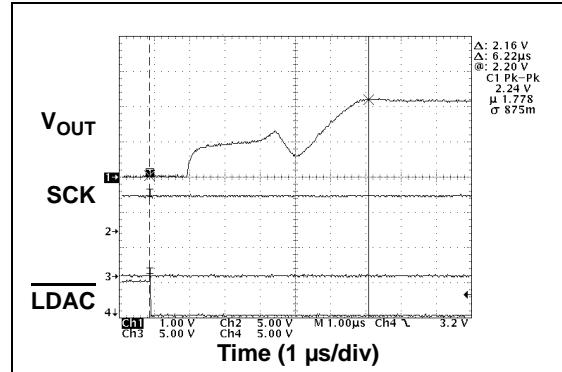


FIGURE 2-32: V_{OUT} Rise Time Exit Shutdown.

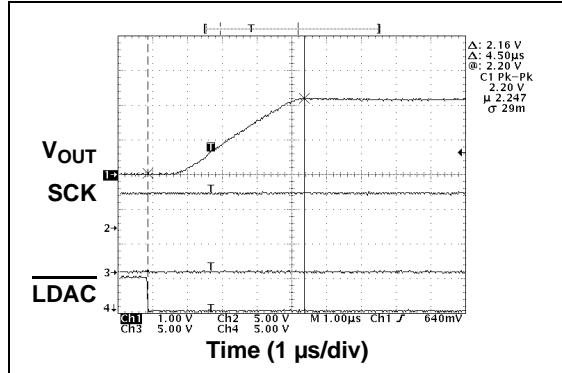


FIGURE 2-30: V_{OUT} Rise Time.

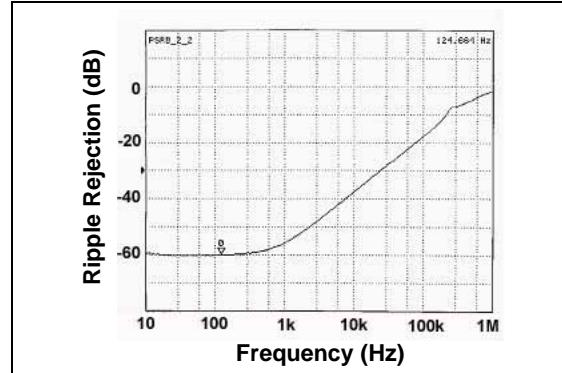


FIGURE 2-33: PSRR vs. Frequency.

3.0 PIN DESCRIPTIONS

The descriptions of the pins are listed in [Table 3-1](#).

TABLE 3-1: PIN FUNCTION TABLE FOR MCP4802/4812/4822

MCP4802/4812/4822	Symbol	Description
MSOP, PDIP, SOIC		
1	V _{DD}	Supply Voltage Input (2.7V to 5.5V)
2	CS	Chip Select Input
3	SCK	Serial Clock Input
4	SDI	Serial Data Input
5	LDAC	Synchronization Input. This pin is used to transfer DAC settings (Input Registers) to the output registers (V _{OUT})
6	V _{OUTB}	DAC _B Output
7	V _{SS}	Ground reference point for all circuitry on the device
8	V _{OUTA}	DAC _A Output

3.1 Supply Voltage Pins (V_{DD}, V_{SS})

V_{DD} is the positive supply voltage input pin. The input supply voltage is relative to V_{SS} and can range from 2.7V to 5.5V. The power supply at the V_{DD} pin should be as clean as possible for a good DAC performance. It is recommended to use an appropriate bypass capacitor of about 0.1 μ F (ceramic) to ground. An additional 10 μ F capacitor (tantalum) in parallel is also recommended to further attenuate high-frequency noise present in application boards.

V_{SS} is the analog ground pin and the current return path of the device. The user must connect the V_{SS} pin to a ground plane through a low-impedance connection. If an analog ground path is available in the application Printed Circuit Board (PCB), it is highly recommended that the V_{SS} pin be tied to the analog ground path or isolated within an analog ground plane of the circuit board.

3.2 Chip Select (CS)

CS is the Chip Select input pin, which requires an active-low to enable serial clock and data functions.

3.3 Serial Clock Input (SCK)

SCK is the SPI compatible serial clock input pin.

3.4 Serial Data Input (SDI)

SDI is the SPI compatible serial data input pin.

3.5 Latch DAC Input (LDAC)

LDAC (latch DAC synchronization input) pin is used to transfer the input latch registers to their corresponding DAC registers (output latches, V_{OUT}). When this pin is low, both V_{OUTA} and V_{OUTB} are updated at the same time with their input register contents. This pin can be tied to low (V_{SS}) if the V_{OUT} update is desired at the rising edge of the CS pin. This pin can be driven by an external control device such as an MCU I/O pin.

3.6 Analog Outputs (V_{OUTA}, V_{OUTB})

V_{OUTA} is the DAC A output pin, and V_{OUTB} is the DAC B output pin. Each output has its own output amplifier. The full-scale range of the DAC output is from V_{SS} to G* V_{REF}, where G is the gain selection option (1x or 2x). The DAC analog output cannot go higher than the supply voltage (V_{DD}).

MCP4802/4812/4822

NOTES:

4.0 GENERAL OVERVIEW

The MCP4802, MCP4812 and MCP4822 are dual voltage output 8-bit, 10-bit and 12-bit DAC devices, respectively. These devices include rail-to-rail output amplifiers, internal voltage reference, shutdown and reset-management circuitry. The devices use an SPI serial communication interface and operate with a single supply voltage from 2.7V to 5.5V.

The DAC input coding of these devices is straight binary. [Equation 4-1](#) shows the DAC analog output voltage calculation.

EQUATION 4-1: ANALOG OUTPUT VOLTAGE (V_{OUT})

$$V_{OUT} = \frac{(2.048V \times D_n)}{2^n} \times G$$

Where:

- 2.048V = Internal voltage reference
- D_n = DAC input code
- G = Gain selection
 - = 2 for \overline{GA} bit = 0
 - = 1 for \overline{GA} bit = 1
- n = DAC Resolution
 - = 8 for MCP4802
 - = 10 for MCP4812
 - = 12 for MCP4822

The ideal output range of each device is:

- **MCP4802 (n = 8)**
 - (a) 0.0V to $255/256 * 2.048V$ when gain setting = 1x.
 - (b) 0.0V to $255/256 * 4.096V$ when gain setting = 2x.
- **MCP4812 (n = 10)**
 - (a) 0.0V to $1023/1024 * 2.048V$ when gain setting = 1x.
 - (b) 0.0V to $1023/1024 * 4.096V$ when gain setting = 2x.
- **MCP4822 (n = 12)**
 - (a) 0.0V to $4095/4096 * 2.048V$ when gain setting = 1x.
 - (b) 0.0V to $4095/4096 * 4.096V$ when gain setting = 2x.

Note: See the output swing voltage specification in [Section 1.0 “Electrical Characteristics”](#).

1 Lsb is the ideal voltage difference between two successive codes. [Table 4-1](#) illustrates the Lsb calculation of each device.

TABLE 4-1: Lsb OF EACH DEVICE

Device	Gain Selection	Lsb Size
MCP4802 (n = 8)	1x	$2.048V/256 = 8 \text{ mV}$
	2x	$4.096V/256 = 16 \text{ mV}$
MCP4812 (n = 10)	1x	$2.048V/1024 = 2 \text{ mV}$
	2x	$4.096V/1024 = 4 \text{ mV}$
MCP4822 (n = 12)	1x	$2.048V/4096 = 0.5 \text{ mV}$
	2x	$4.096V/4096 = 1 \text{ mV}$

4.0.1 INL ACCURACY

Integral Non-Linearity (INL) error for these devices is the maximum deviation between an actual code transition point and its corresponding ideal transition point once offset and gain errors have been removed. The two end points method (from 0x000 to 0xFFFF) is used for the calculation. [Figure 4-1](#) shows the details.

A positive INL error represents transition(s) later than ideal. A negative INL error represents transition(s) earlier than ideal.

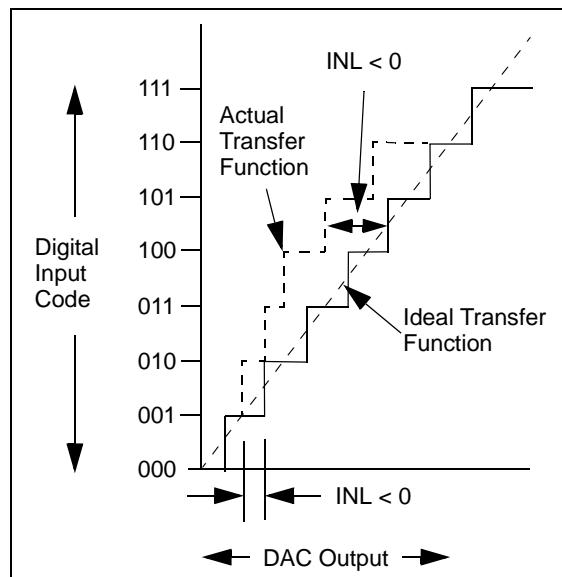


FIGURE 4-1: Example for INL Error.

MCP4802/4812/4822

4.0.2 DNL ACCURACY

A Differential Non-Linearity (DNL) error is the measure of variations in code widths from the ideal code width. A DNL error of zero indicates that every code is exactly 1 LSb wide.

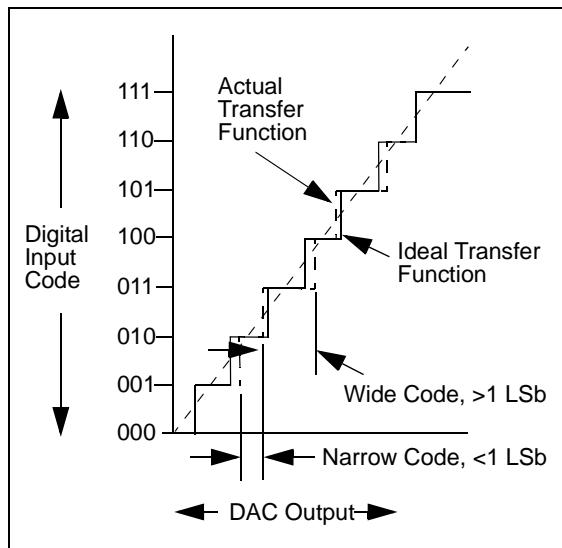


FIGURE 4-2: Example for DNL Error.

4.0.3 OFFSET ERROR

An offset error is the deviation from zero voltage output when the digital input code is zero.

4.0.4 GAIN ERROR

A gain error is the deviation from the ideal output, $V_{REF} - 1$ LSb, excluding the effects of offset error.

4.1 Circuit Descriptions

4.1.1 OUTPUT AMPLIFIERS

The DAC's outputs are buffered with a low-power, precision CMOS amplifier. This amplifier provides low offset voltage and low noise. The output stage enables the device to operate with output voltages close to the power supply rails. Refer to [Section 1.0 “Electrical Characteristics”](#) for the analog output voltage range and load conditions.

In addition to resistive load-driving capability, the amplifier will also drive high capacitive loads without oscillation. The amplifier's strong outputs allow V_{OUT} to be used as a programmable voltage reference in a system.

4.1.1.1 Programmable Gain Block

The rail-to-rail output amplifier has two configurable gain options: a gain of 1x ($\langle GA \rangle = 1$) or a gain of 2x ($\langle GA \rangle = 0$). The default value for this bit is a gain of 2 ($\langle GA \rangle = 0$). This results in an ideal full-scale output of 0.000V to 4.096V due to the internal reference ($V_{REF} = 2.048V$).

4.1.2 VOLTAGE REFERENCE

The MCP4802/4812/4822 devices utilize internal 2.048V voltage reference. The voltage reference has a low temperature coefficient and low noise characteristics. Refer to [Section 1.0 “Electrical Characteristics”](#) for the voltage reference specifications.

4.1.3 POWER-ON RESET CIRCUIT

The internal Power-on Reset (POR) circuit monitors the power supply voltage (V_{DD}) during the device operation. The circuit also ensures that the DAC powers up with high output impedance ($<SHDN> = 0$, typically $500\text{ k}\Omega$). The devices will continue to have a high-impedance output until a valid write command is received and the LDAC pin meets the input low threshold.

If the power supply voltage is less than the POR threshold ($V_{POR} = 2.0\text{V}$, typical), the DACs will be held in their Reset state. The DACs will remain in that state until $V_{DD} > V_{POR}$ and a subsequent write command is received.

Figure 4-3 shows a typical power supply transient pulse and the duration required to cause a reset to occur, as well as the relationship between the duration and trip voltage. A $0.1\text{ }\mu\text{F}$ decoupling capacitor, mounted as close as possible to the V_{DD} pin, can provide additional transient immunity.

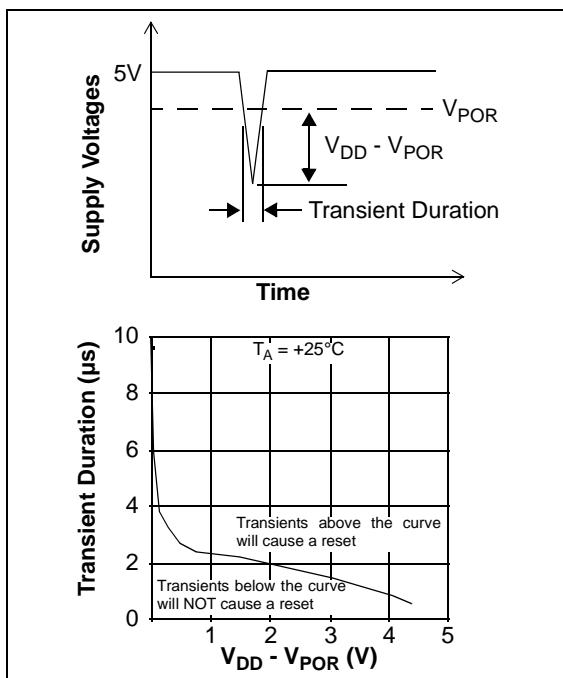


FIGURE 4-3: Typical Transient Response.

4.1.4 SHUTDOWN MODE

The user can shut down each DAC channel selectively using a software command ($<SHDN> = 0$). During Shutdown mode, most of the internal circuits in the channel that was shut down are turned off for power savings. The internal reference is not affected by the shutdown command. The serial interface also remains active, thus allowing a write command to bring the device out of the Shutdown mode. There will be no analog output at the channel that was shut down and the V_{OUT} pin is internally switched to a known resistive load ($500\text{ k}\Omega$, typical). **Figure 4-4** shows the analog output stage during the Shutdown mode.

The device will remain in Shutdown mode until the $<SHDN>$ bit = 1 is latched into the device. When a DAC channel is changed from Shutdown to Active mode, the output settling time takes $< 10\text{ }\mu\text{s}$, but greater than the standard active mode settling time ($4.5\text{ }\mu\text{s}$).

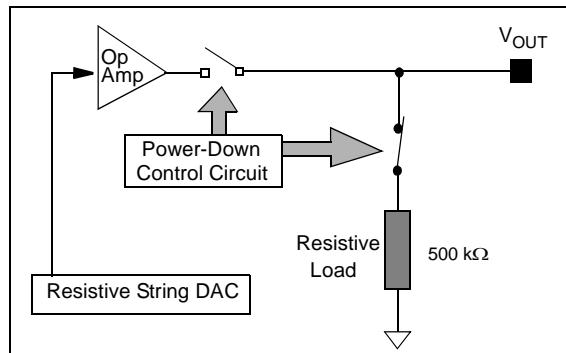


FIGURE 4-4: Output Stage for Shutdown Mode.

MCP4802/4812/4822

NOTES:

5.0 SERIAL INTERFACE

5.1 Overview

The MCP4802/4812/4822 devices are designed to interface directly with the Serial Peripheral Interface (SPI) port, available on many microcontrollers, and supports Mode 0,0 and Mode 1,1. Commands and data are sent to the device via the SDI pin, with data being clocked-in on the rising edge of SCK. The communications are unidirectional and, thus, data cannot be read out of the MCP4802/4812/4822 devices. The \overline{CS} pin must be held low for the duration of a write command. The write command consists of 16 bits and is used to configure the DAC's control and data latches. [Register 5-1](#) to [Register 5-3](#) detail the input register that is used to configure and load the DAC_A and DAC_B registers for each device. [Figure 5-1](#) to [Figure 5-3](#) show the write command for each device.

Refer to [Figure 1-1](#) and SPI Timing Specifications Table for detailed input and output timing specifications for both Mode 0,0 and Mode 1,1 operation.

5.2 Write Command

The write command is initiated by driving the \overline{CS} pin low, followed by clocking the four Configuration bits and the 12 data bits into the SDI pin on the rising edge of SCK. The \overline{CS} pin is then raised, causing the data to be latched into the selected DAC's input registers.

The MCP4802/4812/4822 devices utilize a double-buffered latch structure to allow both DAC_A 's and DAC_B 's outputs to be synchronized with the \overline{LDAC} pin, if desired.

By bringing down the \overline{LDAC} pin to a low state, the contents stored in the DAC's input registers are transferred into the DAC's output registers (V_{OUT}), and both V_{OUTA} and V_{OUTB} are updated at the same time.

All writes to the MCP4802/4812/4822 devices are 16-bit words. Any clocks after the first 16th clock will be ignored. The Most Significant four bits are Configuration bits. The remaining 12 bits are data bits. No data can be transferred into the device with \overline{CS} high. The data transfer will only occur if 16 clocks have been transferred into the device. If the rising edge of \overline{CS} occurs prior, shifting of data into the input registers will be aborted.

MCP4802/4812/4822

REGISTER 5-1: WRITE COMMAND REGISTER FOR MCP4822 (12-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
Ā/B	—	GA	SHDN	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0		
bit 15								bit 0									

REGISTER 5-2: WRITE COMMAND REGISTER FOR MCP4812 (10-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
Ā/B	—	GA	SHDN	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	x	x		
bit 15								bit 0									

REGISTER 5-3: WRITE COMMAND REGISTER FOR MCP4802 (8-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
Ā/B	—	GA	SHDN	D7	D6	D5	D4	D3	D2	D1	D0	x	x	x	x	x	x
bit 15								bit 0									

Where:

bit 15 **Ā/B:** DAC_A or DAC_B Selection bit

1 = Write to DAC_B
0 = Write to DAC_A

bit 14 — Don't Care

bit 13 **GA:** Output Gain Selection bit

1 = 1x ($V_{OUT} = V_{REF} * D/4096$)
0 = 2x ($V_{OUT} = 2 * V_{REF} * D/4096$), where internal $V_{REF} = 2.048V$.

bit 12 **SHDN:** Output Shutdown Control bit

1 = Active mode operation. V_{OUT} is available.
0 = Shutdown the selected DAC channel. Analog output is not available at the channel that was shut down.
 V_{OUT} pin is connected to 500 kΩ (typical).

bit 11-0 **D11:D0:** DAC Input Data bits. Bit x is ignored.

Legend

R = Readable bit
-n = Value at POR

W = Writable bit
1 = bit is set

U = Unimplemented bit, read as '0'
0 = bit is cleared
x = bit is unknown

MCP4802/4812/4822

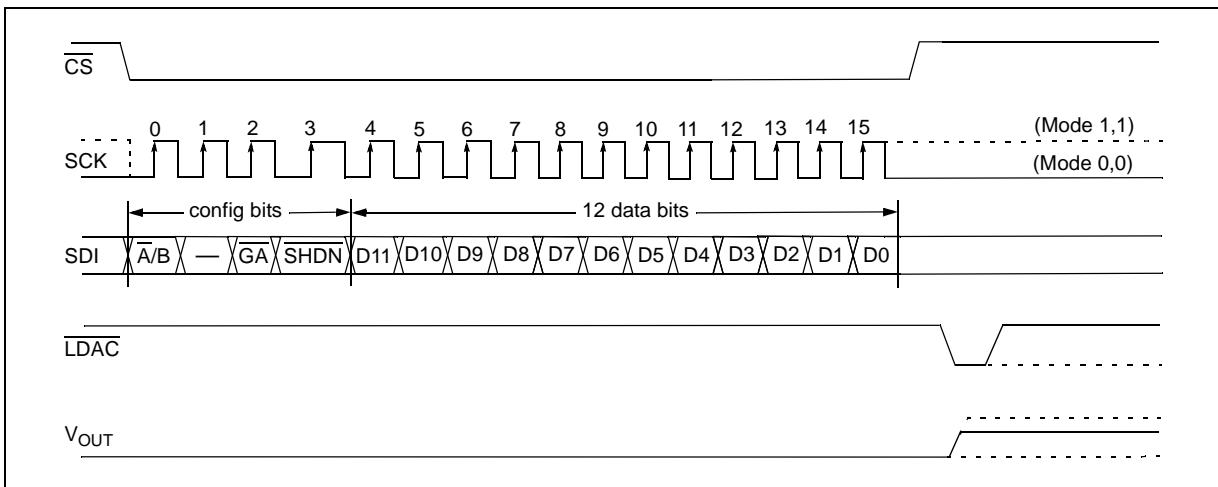


FIGURE 5-1: Write Command for MCP4822 (12-bit DAC).

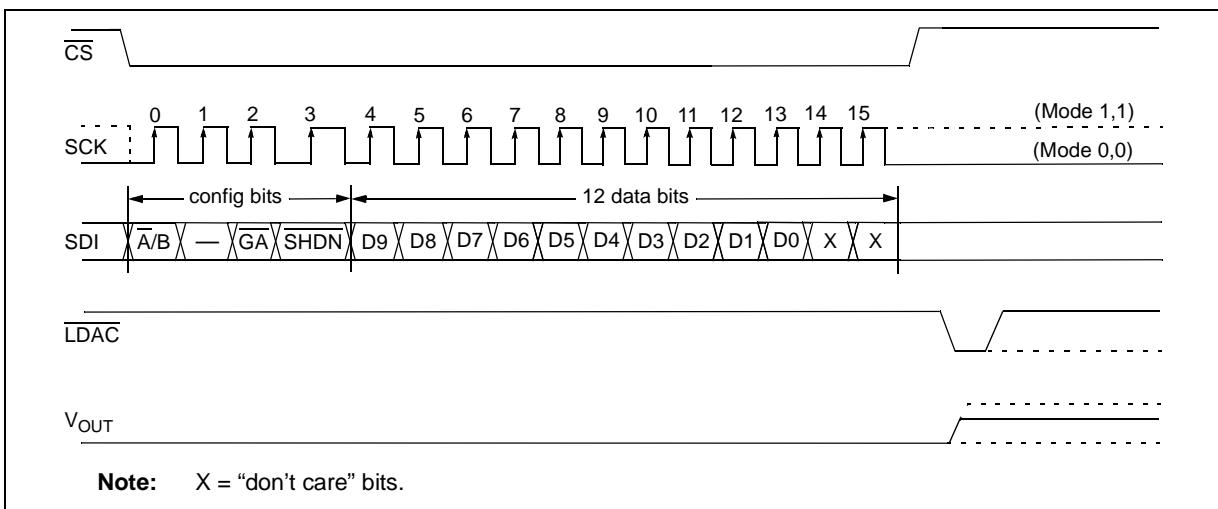


FIGURE 5-2: Write Command for MCP4812 (10-bit DAC).

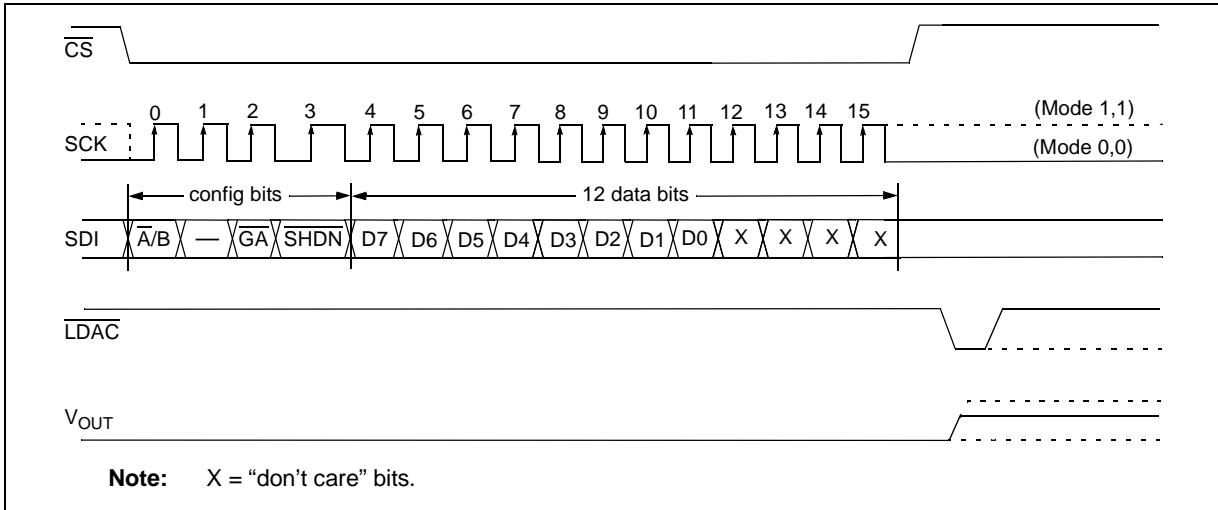


FIGURE 5-3: Write Command for MCP4802 (8-bit DAC).

MCP4802/4812/4822

NOTES:

6.0 TYPICAL APPLICATIONS

The MCP4802/4812/4822 family of devices are general purpose DACs for various applications where a precision operation with low-power and internal voltage reference is required.

Applications generally suited for the devices are:

- Set Point or Offset Trimming
- Sensor Calibration
- Precision Selectable Voltage Reference
- Portable Instrumentation (Battery-Powered)
- Calibration of Optical Communication Devices

6.1 Digital Interface

The MCP4802/4812/4822 devices utilize a 3-wire synchronous serial protocol to transfer the DAC's setup and input codes from the digital devices. The serial protocol can be interfaced to SPI or Microwire peripherals that is common on many microcontroller units (MCUs), including Microchip's PIC® MCUs and dsPIC® DSCs.

In addition to the three serial connections (\overline{CS} , SCK and SDI), the LDAC signal synchronizes the two DAC outputs. By bringing down the LDAC pin to "low", all DAC input codes and settings in the two DAC input registers are latched into their DAC output registers at the same time. Therefore, both DAC_A and DAC_B outputs are updated at the same time. [Figure 6-1](#) shows an example of the pin connections. Note that the LDAC pin can be tied low (V_{SS}) to reduce the required connections from four to three I/O pins. In this case, the DAC output can be immediately updated when a valid 16 clock transmission has been received and the CS pin has been raised.

6.2 Power Supply Considerations

The typical application will require a bypass capacitor in order to filter out the noise in the power supply traces. The noise can be induced onto the power supply's traces from various events such as digital switching or as a result of changes on the DAC's output. The bypass capacitor helps to minimize the effect of these noise sources. [Figure 6-1](#) illustrates an appropriate bypass strategy. In this example, two bypass capacitors are used in parallel: (a) 0.1 μF (ceramic) and (b) 10 μF (tantalum). These capacitors should be placed as close to the device power pin (V_{DD}) as possible (within 4 mm).

The power source supplying these devices should be as clean as possible. If the application circuit has separate digital and analog power supplies, V_{DD} and V_{SS} of the device should reside on the analog plane.

6.3 Output Noise Considerations

The voltage noise density (in $\mu V/\sqrt{Hz}$) is illustrated in [Figure 2-13](#). This noise appears at V_{OUTX} , and is primarily a result of the internal reference voltage. Its 1/f corner (f_{CORNER}) is approximately 400 Hz.

[Figure 2-14](#) illustrates the voltage noise (in mV_{RMS} or mV_{P-P}). A small bypass capacitor on V_{OUTX} is an effective method to produce a single-pole Low-Pass Filter (LPF) that will reduce this noise. For instance, a bypass capacitor sized to produce a 1 kHz LPF would result in an ENREF of about 100 μV_{RMS} . This would be necessary when trying to achieve the low DNL error performance (at $G = 1$) that the MCP4802/4812/4822 devices are capable of. The tested range for stability is .001 μF through 4.7 μF .

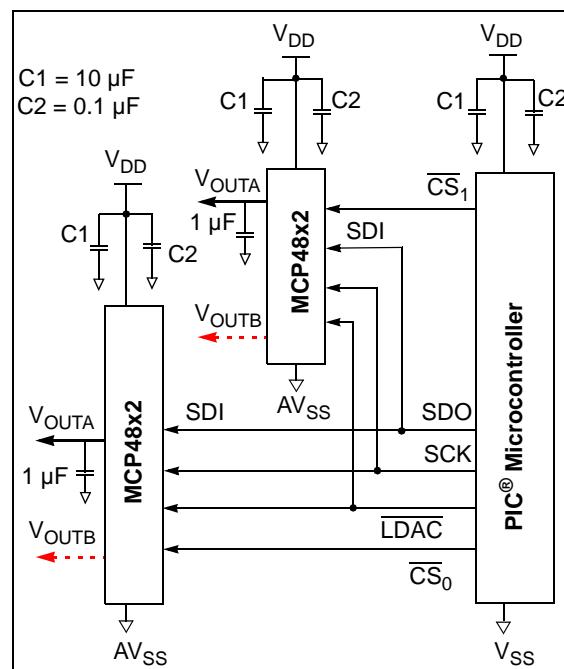


FIGURE 6-1: Typical Connection Diagram.

6.4 Layout Considerations

Inductively-coupled AC transients and digital switching noises can degrade the output signal integrity, and potentially reduce the device performance. Careful board layout will minimize these effects and increase the Signal-to-Noise Ratio (SNR). Bench testing has shown that a multi-layer board utilizing a low-inductance ground plane, isolated inputs and isolated outputs with proper decoupling, is critical for the best performance. Particularly harsh environments may require shielding of critical signals.

Breadboards and wire-wrapped boards are not recommended if low noise is desired.

MCP4802/4812/4822

6.5 Single-Supply Operation

The MCP4802/4812/4822 family of devices are rail-to-rail voltage output DAC devices designed to operate with a V_{DD} range of 2.7V to 5.5V. Its output amplifier is robust enough to drive small-signal loads directly. Therefore, it does not require any external output buffer for most applications.

6.5.1 DC SET POINT OR CALIBRATION

A common application for the devices is a digitally-controlled set point and/or calibration of variable parameters, such as sensor offset or slope. For example, the MCP4822 provides 4096 output steps. If $G = 1$ is selected, the internal 2.048V V_{REF} would produce 500 μ V of resolution. If $G = 2$ is selected, the internal 2.048V V_{REF} would produce 1 mV of resolution.

6.5.1.1 Decreasing Output Step Size

If the application is calibrating the bias voltage of a diode or transistor, a bias voltage range of 0.8V may be desired with about 200 μ V resolution per step. Two common methods to achieve a 0.8V range are to either reduce V_{REF} to 0.82V (using the MCP49XX family device that uses external reference) or use a voltage divider on the DAC's output.

Using a V_{REF} is an option if the V_{REF} is available with the desired output voltage range. However, occasionally, when using a low-voltage V_{REF} , the noise floor causes SNR error that is intolerable. Using a voltage divider method is another option and provides some advantages when V_{REF} needs to be very low or when the desired output voltage is not available. In this case, a larger value V_{REF} is used while two resistors scale the output range down to the precise desired level.

[Example 6-1](#) illustrates this concept. Note that the bypass capacitor on the output of the voltage divider plays a critical function in attenuating the output noise of the DAC and the induced noise from the environment.

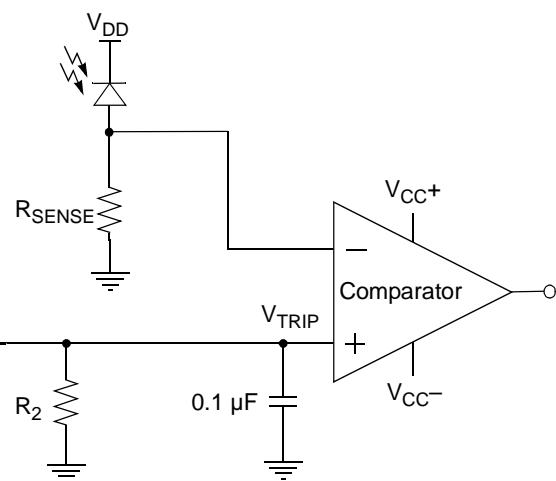
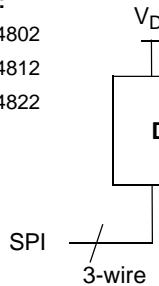
EXAMPLE 6-1: EXAMPLE CIRCUIT OF SET POINT OR THRESHOLD CALIBRATION

(a) Single Output DAC:

MCP4801
MCP4811
MCP4821

(b) Dual Output DAC:

MCP4802
MCP4812
MCP4822



$$V_{OUT} = 2.048 \cdot G \cdot \frac{D_n}{2^N}$$

$$V_{trip} = V_{OUT} \left(\frac{R_2}{R_1 + R_2} \right)$$

G	= Gain selection (1x or 2x)
D _n	= Digital value of DAC (0-255) for MCP4801/MCP4802
=	= Digital value of DAC (0-1023) for MCP4811/MCP4812
=	= Digital value of DAC (0-4095) for MCP4821/MCP4822
N	= DAC bit resolution

MCP4802/4812/4822

6.5.1.2 Building a “Window” DAC

When calibrating a set point or threshold of a sensor, typically only a small portion of the DAC output range is utilized. If the LSb size is adequate enough to meet the application's accuracy needs, the unused range is sacrificed without consequences. If greater accuracy is needed, then the output range will need to be reduced to increase the resolution around the desired threshold.

If the threshold is not near V_{REF} , $2V_{REF}$ or V_{SS} , then creating a “window” around the threshold has several advantages. One simple method to create this “window” is to use a voltage divider network with a pull-up and pull-down resistor. [Example 6-2](#) shows this concept.

EXAMPLE 6-2: SINGLE-SUPPLY “WINDOW” DAC

(a) Single Output DAC:

MCP4801

MCP4811

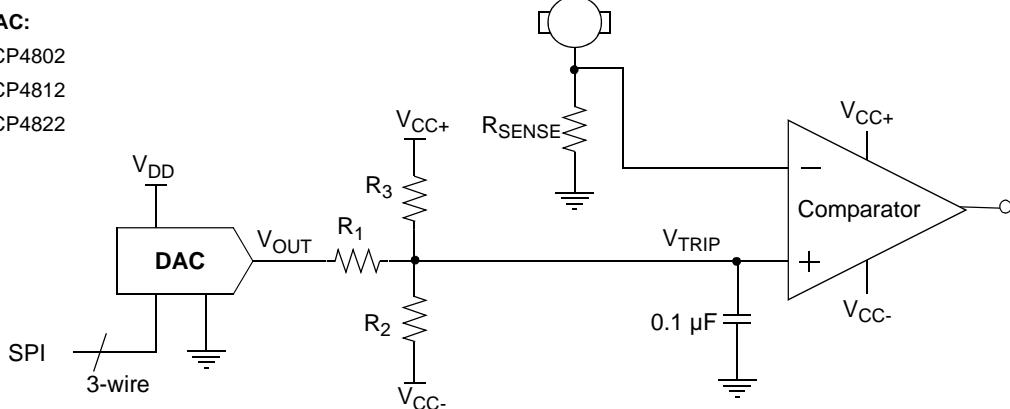
MCP4821

(b) Dual Output DAC:

MCP4802

MCP4812

MCP4822



$$V_{OUT} = 2.048 \cdot G \cdot \frac{D_n}{2^N}$$

G = Gain selection (1x or 2x)

D_n = Digital value of DAC (0-255) for MCP4801/MCP4802

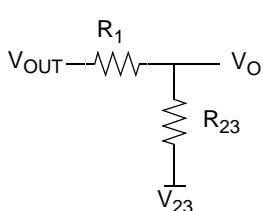
= Digital value of DAC (0-1023) for MCP4811/MCP4812

= Digital value of DAC (0-4095) for MCP4821/MCP4822

N = DAC bit resolution

Thevenin Equivalent

$$\left\{ \begin{array}{l} R_{23} = \frac{R_2 R_3}{R_2 + R_3} \\ V_{23} = \frac{(V_{CC+} R_2) + (V_{CC-} R_3)}{R_2 + R_3} \\ V_{trip} = \frac{V_{OUT} R_{23} + V_{23} R_1}{R_1 + R_{23}} \end{array} \right.$$



MCP4802/4812/4822

6.6 Bipolar Operation

Bipolar operation is achievable using the MCP4802/4812/4822 family of devices by utilizing an external operational amplifier (op amp). This configuration is desirable due to the wide variety and availability of op amps. This allows a general purpose DAC, with its cost and availability advantages, to meet almost any desired output voltage range, power and noise performance.

Example 6-3 illustrates a simple bipolar voltage source configuration. R_1 and R_2 allow the gain to be selected, while R_3 and R_4 shift the DAC's output to a selected offset. Note that R_4 can be tied to V_{DD} , instead of V_{SS} , if a higher offset is desired. Also note that a pull-up to V_{DD} could be used instead of R_4 , or in addition to R_4 , if a higher offset is desired.

EXAMPLE 6-3: DIGITALLY-CONTROLLED BIPOLAR VOLTAGE SOURCE

(a) Single Output DAC:

MCP4801

MCP4811

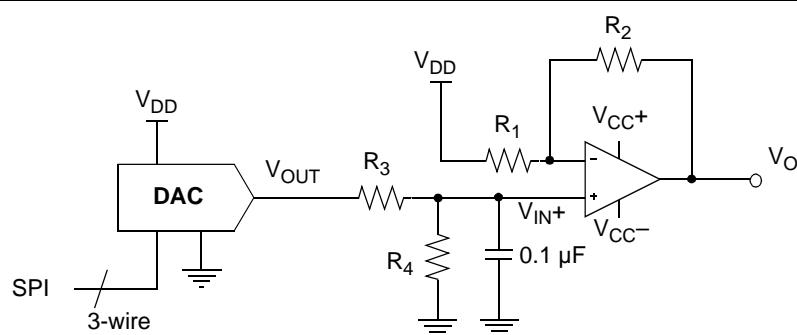
MCP4821

(b) Dual Output DAC:

MCP4802

MCP4812

MCP4822



$$V_{OUT} = 2.048 \cdot G \cdot \frac{D_n}{2^N}$$

$$V_{IN+} = \frac{V_{OUT}R_4}{R_3 + R_4}$$

$$V_O = V_{IN+} \left(1 + \frac{R_2}{R_1} \right) - V_{DD} \left(\frac{R_2}{R_1} \right)$$

G = Gain selection (1x or 2x)

D_n = Digital value of DAC (0-255) for MCP4801/MCP4802

= Digital value of DAC (0-1023) for MCP4811/MCP4812

= Digital value of DAC (0-4095) for MCP4821/MCP4822

N = DAC bit resolution

6.6.1 DESIGN EXAMPLE: DESIGN A BIPOLAR DAC USING [Example 6-3](#) WITH 12-BIT MCP4822 OR MCP4821

An output step magnitude of 1 mV, with an output range of $\pm 2.05V$, is desired for a particular application.

Step 1: Calculate the range: $+2.05V - (-2.05V) = 4.1V$.

Step 2: Calculate the resolution needed:

$$4.1V / 1 \text{ mV} = 4100$$

Since $2^{12} = 4096$, 12-bit resolution is desired.

Step 3: The amplifier gain (R_2/R_1), multiplied by full-scale V_{OUT} (4.096V), must be equal to the desired minimum output to achieve bipolar operation. Since any gain can be realized by choosing resistor values (R_1+R_2), the V_{REF} value must be selected first. If a V_{REF} of 4.096V is used (G=2), solve for the amplifier's gain by setting the DAC to 0, knowing that the output needs to be -2.05V.

The equation can be simplified to:

$$\frac{-R_2}{R_1} = \frac{-2.05}{4.096V} \quad \frac{R_2}{R_1} = \frac{1}{2}$$

If $R_1 = 20 \text{ k}\Omega$ and $R_2 = 10 \text{ k}\Omega$, the gain will be 0.5.

Step 4: Next, solve for R_3 and R_4 by setting the DAC to 4096, knowing that the output needs to be +2.05V.

$$\frac{R_4}{(R_3 + R_4)} = \frac{2.05V + (0.5 \cdot 4.096V)}{1.5 \cdot 4.096V} = \frac{2}{3}$$

If $R_4 = 20 \text{ k}\Omega$, then $R_3 = 10 \text{ k}\Omega$

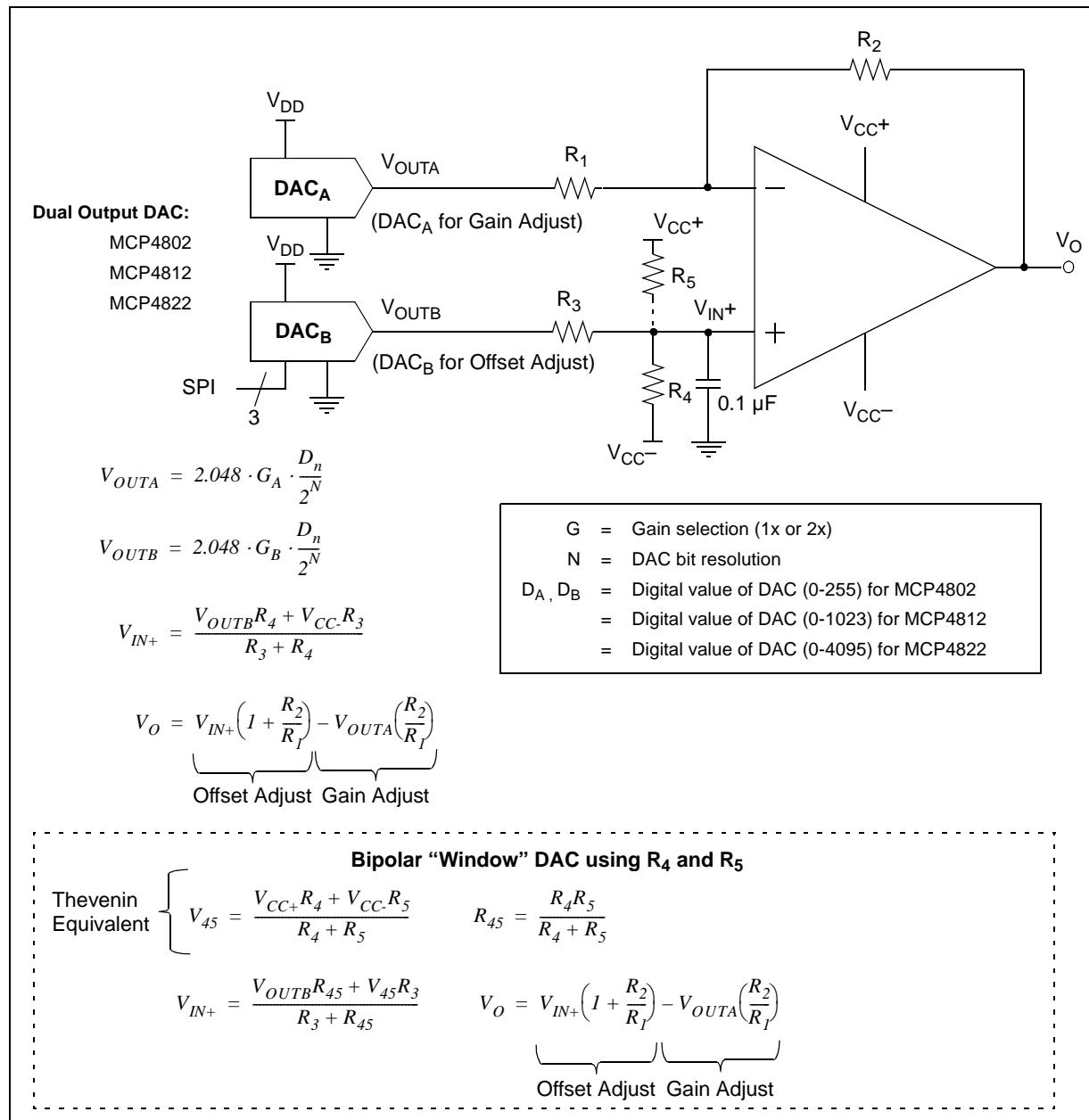
6.7 Selectable Gain and Offset Bipolar Voltage Output Using a Dual Output DAC

In some applications, precision digital control of the output range is desirable. **Example 6-4** illustrates how to use the MCP4802/4812/4822 family of devices to achieve this in a bipolar or single-supply application.

This circuit is typically used for linearizing a sensor whose slope and offset varies.

The equation to design a bipolar “window” DAC would be utilized if R_3 , R_4 and R_5 are populated.

EXAMPLE 6-4: BIPOLAR VOLTAGE SOURCE WITH SELECTABLE GAIN AND OFFSET



MCP4802/4812/4822

6.8 Designing a Double-Precision DAC Using a Dual DAC

Example 6-5 illustrates how to design a single-supply voltage output capable of up to 24-bit resolution from a dual 12-bit DAC (MCP4822). This design is simply a voltage divider with a buffered output.

As an example, if an application similar to the one developed in [Section 6.6.1 “Design Example: Design a Bipolar DAC Using Example 6-3 with 12-bit MCP4822 or MCP4821”](#) required a resolution of 1 μ V instead of 1 mV, and a range of 0V to 4.1V, then 12-bit resolution would not be adequate.

Step 1: Calculate the resolution needed:

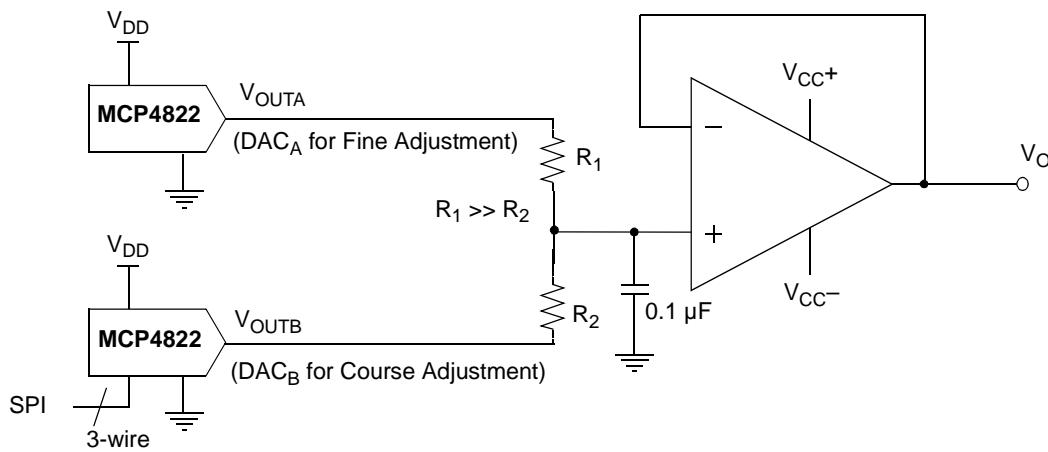
$4.1\text{V}/1\text{ }\mu\text{V} = 4.1 \times 10^6$. Since $2^{22} = 4.2 \times 10^6$, 22-bit resolution is desired. Since DNL = ± 0.75 LSb, this design can be done with the 12-bit MCP4822 DAC.

Step 2: Since DAC_B’s V_{OUTB} has a resolution of 1 mV, its output only needs to be “pulled” 1/1000 to meet the 1 μ V target. Dividing V_{OUTA} by 1000 would allow the application to compensate for DAC_B’s DNL error.

Step 3: If R_2 is 100 Ω , then R_1 needs to be 100 k Ω .

Step 4: The resulting transfer function is shown in the equation of [Example 6-5](#).

EXAMPLE 6-5: SIMPLE, DOUBLE-PRECISION DAC WITH MCP4822



$$V_{OUTA} = 2.048 \cdot G_A \cdot \frac{D_A}{2^{12}}$$

$$V_{OUTB} = 2.048 \cdot G_B \cdot \frac{D_B}{2^{12}}$$

$$V_O = \frac{V_{OUTA}R_2 + V_{OUTB}R_1}{R_1 + R_2}$$

G_x = Gain selection (1x or 2x)

D_n = Digital value of DAC (0-4096)

6.9 Building Programmable Current Source

[Example 6-6](#) shows an example of building a programmable current source using a voltage follower. The current sensor (sensor resistor) is used to convert the DAC voltage output into a digitally-selectable current source.

Adding the resistor network from [Example 6-2](#) would be advantageous in this application. The smaller R_{SENSE} is, the less power dissipated across it.

However, this also reduces the resolution that the current can be controlled with. The voltage divider, or “window”, DAC configuration would allow the range to be reduced, thus increasing resolution around the range of interest. When working with very small sensor voltages, plan on eliminating the amplifier’s offset error by storing the DAC’s setting under known sensor conditions.

EXAMPLE 6-6: DIGITALLY-CONTROLLED CURRENT SOURCE

(a) Single Output DAC:

- MCP4801
- MCP4811
- MCP4821

(b) Dual Output DAC:

- MCP4802
- MCP4812
- MCP4822

$$I_b = \frac{I_L}{\beta}$$

$$I_L = \frac{V_{OUT}}{R_{sense}} \times \frac{\beta}{\beta + 1}$$

where β = Common-Emitter Current Gain.

G	= Gain selection (1x or 2x)
D _n	= Digital value of DAC (0-255) for MCP4801/MCP4802
=	= Digital value of DAC (0-1023) for MCP4811/MCP4812
=	= Digital value of DAC (0-4095) for MCP4821/MCP4822
N	= DAC bit resolution

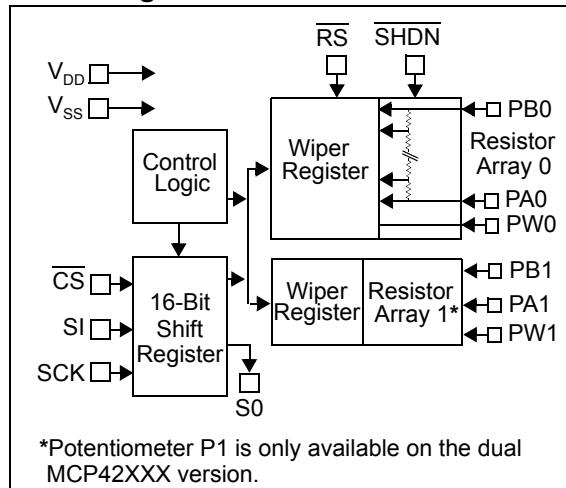
**MICROCHIP****MCP41XXX/42XXX**

Single/Dual Digital Potentiometer with SPI™ Interface

Features

- 256 taps for each potentiometer
- Potentiometer values for 10 kΩ, 50 kΩ and 100 kΩ
- Single and dual versions
- SPI™ serial interface (mode 0,0 and 1,1)
- ±1 LSB max INL & DNL
- Low power CMOS technology
- 1 µA maximum supply current in static operation
- Multiple devices can be daisy-chained together (MCP42XXX only)
- Shutdown feature open circuits of all resistors for maximum power savings
- Hardware shutdown pin available on MCP42XXX only
- Single supply operation (2.7V - 5.5V)
- Industrial temperature range: -40°C to +85°C
- Extended temperature range: -40°C to +125°C

Block Diagram



*Potentiometer P1 is only available on the dual MCP42XXX version.

Description

The MCP41XXX and MCP42XXX devices are 256-position, digital potentiometers available in 10 kΩ, 50 kΩ and 100 kΩ resistance versions. The MCP41XXX is a single-channel device and is offered in an 8-pin PDIP or SOIC package. The MCP42XXX contains two independent channels in a 14-pin PDIP, SOIC or TSSOP package. The wiper position of the MCP41XXX/42XXX varies linearly and is controlled via an industry-standard SPI interface. The devices consume <1 µA during static operation. A software shutdown feature is provided that disconnects the "A" terminal from the resistor stack and simultaneously connects the wiper to the "B" terminal. In addition, the dual MCP42XXX has a SHDN pin that performs the same function in hardware. During shutdown mode, the contents of the wiper register can be changed and the potentiometer returns from shutdown to the new value. The wiper is reset to the mid-scale position (80h) upon power-up. The RS (reset) pin implements a hardware reset and also returns the wiper to mid-scale. The MCP42XXX SPI interface includes both the SI and SO pins, allowing daisy-chaining of multiple devices. Channel-to-channel resistance matching on the MCP42XXX varies by less than 1%. These devices operate from a single 2.7 - 5.5V supply and are specified over the extended and industrial temperature ranges.

Package Types

PDIP/SOIC

CS	1	8	V _{DD}
SCK	2	7	PB0
SI	3	6	PW0
V _{SS}	4	5	PA0

PDIP/SOIC/TSSOP

CS	1	14	V _{DD}
SCK	2	13	SO
SI	3	12	SHDN
V _{SS}	4	11	RS
PB1	5	10	PB0
PW1	6	9	PW0
PA1	7	8	PA0

MCP41XXX/42XXX

1.0 ELECTRICAL CHARACTERISTICS

DC CHARACTERISTICS: 10 kΩ VERSION

Electrical Characteristics: Unless otherwise indicated, $V_{DD} = +2.7V$ to $5.5V$, $T_A = -40^\circ C$ to $+85^\circ C$ (TSSOP devices are only specified at $+25^\circ C$ and $+85^\circ C$). Typical specifications represent values for $V_{DD} = 5V$, $V_{SS} = 0V$, $V_B = 0V$, $T_A = +25^\circ C$.						
Parameters	Sym	Min	Typ	Max	Units	Conditions
Rheostat Mode						
Nominal Resistance	R	8	10	12	kΩ	$T_A = +25^\circ C$ (Note 1)
Rheostat Differential Non Linearity	R-DNL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Integral Non Linearity	R-INL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Tempco	$\Delta R_{AB}/\Delta T$	—	800	—	ppm/°C	
Wiper Resistance	R_W	—	52	100	Ω	$V_{DD} = 5.5V$, $I_W = 1$ mA, code 00h
	R_W	—	73	125	Ω	$V_{DD} = 2.7V$, $I_W = 1$ mA, code 00h
Wiper Current	I_W	-1	—	+1	mA	
Nominal Resistance Match	$\Delta R/R$	—	0.2	1	%	MCP42010 only , P0 to P1; $T_A = +25^\circ C$
Potentiometer Divider						
Resolution	N	8	—	—	Bits	
Monotonicity	N	8	—	—	Bits	
Differential Non-Linearity	DNL	-1	$\pm 1/4$	+1	LSB	Note 3
Integral Non-Linearity	INL	-1	$\pm 1/4$	+1	LSB	Note 3
Voltage Divider Tempco	$\Delta V_W/\Delta T$	—	1	—	ppm/°C	Code 80h
Full Scale Error	V_{WFSE}	-2	-0.7	0	LSB	Code FFh, $V_{DD} = 5V$, see Figure 2-25
	V_{WFSE}	-2	-0.7	0	LSB	Code FFh, $V_{DD} = 3V$, see Figure 2-25
Zero Scale Error	V_{WZSE}	0	+0.7	+2	LSB	Code 00h, $V_{DD} = 5V$, see Figure 2-25
	V_{WZSE}	0	+0.7	+2	LSB	Code 00h, $V_{DD} = 3V$, see Figure 2-25
Resistor Terminals						
Voltage Range	$V_{A,B,W}$	0	—	V_{DD}	V	Note 4
Capacitance (C_A or C_B)		—	15	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Capacitance	C_W	—	5.6	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Dynamic Characteristics (All dynamic characteristics use $V_{DD} = 5V$)						
Bandwidth -3dB	BW	—	1	—	MHz	$V_B = 0V$, Measured at Code 80h, Output Load = 30 pF
Settling Time	t_S	—	2	—	μS	$V_A = V_{DD}$, $V_B = 0V$, $\pm 1\%$ Error Band, Transition from Code 00h to Code 80h, Output Load = 30 pF
Resistor Noise Voltage	e_{NWB}	—	9	—	nV/√Hz	$V_A = \text{Open}$, Code 80h, $f = 1$ kHz
Crosstalk	C_T	—	-95	—	dB	$V_A = V_{DD}$, $V_B = 0V$ (Note 5)
Digital Inputs/Outputs (CS, SCK, SI, SO) See Figure 2-12 for RS and SHDN pin operation						
Schmitt Trigger High-Level Input Voltage	V_{IH}	0.7 V_{DD}	—	—	V	
Schmitt Trigger Low-Level Input Voltage	V_{IL}	—	—	0.3 V_{DD}	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	0.05 V_{DD}	—		
Low-Level Output Voltage	V_{OL}	—	—	0.40	V	$I_{OL} = 2.1$ mA, $V_{DD} = 5V$
High-Level Output Voltage	V_{OH}	$V_{DD} - 0.5$	—	—	V	$I_{OH} = -400$ μA, $V_{DD} = 5V$
Input Leakage Current	I_{LI}	-1	—	+1	μA	$\overline{CS} = V_{DD}$, $V_{IN} = V_{SS}$ or V_{DD} , includes V_A SHDN=0
Pin Capacitance (All inputs/outputs)	C_{IN} , C_{OUT}	—	10	—	pF	$V_{DD} = 5.0V$, $T_A = +25^\circ C$, $f_c = 1$ MHz
Power Requirements						
Operating Voltage Range	V_{DD}	2.7	—	5.5	V	
Supply Current, Active	I_{DDA}	—	340	500	μA	$V_{DD} = 5.5V$, $\overline{CS} = V_{SS}$, $f_{SCK} = 10$ MHz, SO = Open, Code FFh (Note 6)
Supply Current, Static	I_{DPS}	—	0.01	1	μA	\overline{CS} , \overline{SHDN} , $\overline{RS} = V_{DD} = 5.5V$, SO = Open (Note 6)
Power Supply Sensitivity	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 4.5V - 5.5V$, $V_A = 4.5V$, Code 80h
	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 2.7V - 3.3V$, $V_A = 2.7V$, Code 80h

- Note 1:** $V_{AB} = V_{DD}$, no connection on wiper.
- 2:** Rheostat position non-linearity R-INL is the deviation from an ideal value measured between the maximum resistance and the minimum resistance wiper positions. R-DNL measures the relative step change from the ideal between successive tap positions. $I_W = 50$ μA for $V_{DD} = 3V$ and $I_W = 400$ μA for $V_{DD} = 5V$ for 10 kΩ version. See Figure 2-26 for test circuit.
- 3:** INL and DNL are measured at V_W with the device configured in the voltage divider or potentiometer mode. $V_A = V_{DD}$ and $V_B = 0V$. DNL specification limits of ± 1 LSB max are specified monotonic operating conditions. See Figure 2-25 for test circuit.
- 4:** Resistor terminals A,B and W have no restrictions on polarity with respect to each other. Full-scale and zero-scale error were measured using Figure 2-25.
- 5:** Measured at V_W pin where the voltage on the adjacent V_W pin is swinging full-scale.
- 6:** Supply current is independent of current through the potentiometers.

DC CHARACTERISTICS: 50 kΩ VERSION

Electrical Characteristics: Unless otherwise indicated, $V_{DD} = +2.7V$ to $5.5V$, $T_A = -40^\circ C$ to $+85^\circ C$ (TSSOP devices are only specified at $+25^\circ C$ and $+85^\circ C$). Typical specifications represent values for $V_{DD} = 5V$, $V_{SS} = 0V$, $V_B = 0V$, $T_A = +25^\circ C$.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Rheostat Mode						
Nominal Resistance	R	35	50	65	kΩ	$T_A = +25^\circ C$ (Note 1)
Rheostat Differential Non-Linearity	R-DNL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Integral Non-Linearity	R-INL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Tempco	$\Delta R_{AB}/\Delta T$	—	800	—	ppm/ $^\circ C$	
Wiper Resistance	R_W	—	125	175	Ω	$V_{DD} = 5.5V$, $I_W = 1$ mA, code 00h
	R_W	—	175	250	Ω	$V_{DD} = 2.7V$, $I_W = 1$ mA, code 00h
Wiper Current	I_W	-1	—	+1	mA	
Nominal Resistance Match	$\Delta R/R$	—	0.2	1	%	MCP42050 only , P0 to P1; $T_A = +25^\circ C$
Potentiometer Divider						
Resolution	N	8	—	—	Bits	
Monotonicity	N	8	—	—	Bits	
Differential Non-Linearity	DNL	-1	$\pm 1/4$	+1	LSB	Note 3
Integral Non-Linearity	INL	-1	$\pm 1/4$	+1	LSB	Note 3
Voltage Divider Tempco	$\Delta V_W/\Delta T$	—	1	—	ppm/ $^\circ C$	Code 80h
Full-Scale Error	V_{WFSE}	-1	-0.25	0	LSB	Code FFh, $V_{DD} = 5V$, see Figure 2-25
	V_{WFSE}	-1	-0.35	0	LSB	Code FFh, $V_{DD} = 3V$, see Figure 2-25
Zero-Scale Error	V_{WZSE}	0	+0.25	+1	LSB	Code 00h, $V_{DD} = 5V$, see Figure 2-25
	V_{WZSE}	0	+0.35	+1	LSB	Code 00h, $V_{DD} = 3V$, see Figure 2-25
Resistor Terminals						
Voltage Range	$V_{A,B,W}$	0	—	V_{DD}		Note 4
Capacitance (C_A or C_B)		—	11	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Capacitance	C_W	—	5.6	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Dynamic Characteristics (All dynamic characteristics use $V_{DD} = 5V$)						
Bandwidth -3dB	BW	—	280	—	MHz	$V_B = 0V$, Measured at Code 80h, Output Load = 30 pF
Settling Time	t_S	—	8	—	μS	$V_A = V_{DD}$, $V_B = 0V$, $\pm 1\%$ Error Band, Transition from Code 00h to Code 80h, Output Load = 30 pF
Resistor Noise Voltage	e_{NWB}	—	20	—	nV/ \sqrt{Hz}	$V_A = \text{Open}$, Code 80h, $f = 1$ kHz
Crosstalk	C_T	—	-95	—	dB	$V_A = V_{DD}$, $V_B = 0V$ (Note 5)
Digital Inputs/Outputs (CS, SCK, SI, SO) See Figure 2-12 for RS and SHDN pin operation.						
Schmitt Trigger High-Level Input Voltage	V_{IH}	0.7 V_{DD}	—	—	V	
Schmitt Trigger Low-Level Input Voltage	V_{IL}	—	—	0.3 V_{DD}	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	0.05 V_{DD}	—		
Low-Level Output Voltage	V_{OL}	—	—	0.40	V	$I_{OL} = 2.1$ mA, $V_{DD} = 5V$
High-Level Output Voltage	V_{OH}	$V_{DD} - 0.5$	—	—	V	$I_{OH} = -400$ μA, $V_{DD} = 5V$
Input Leakage Current	I_{LI}	-1	—	+1	μA	$CS = V_{DD}$, $V_{IN} = V_{SS}$ or V_{DD} , includes V_A SHDN=0
Pin Capacitance (All inputs/outputs)	C_{IN}, C_{OUT}	—	10	—	pF	$V_{DD} = 5.0V$, $T_A = +25^\circ C$, $f_c = 1$ MHz
Power Requirements						
Operating Voltage Range	V_{DD}	2.7	—	5.5	V	
Supply Current, Active	I_{DDA}	—	340	500	μA	$V_{DD} = 5.5V$, $CS = V_{SS}$, $f_{SCK} = 10$ MHz, SO = Open, Code FFh (Note 6)
Supply Current, Static	I_{DDS}	—	0.01	1	μA	CS , SHDN, RS = $V_{DD} = 5.5V$, SO = Open (Note 6)
Power Supply Sensitivity	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 4.5V - 5.5V$, $V_A = 4.5V$, Code 80h
	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 2.7V - 3.3V$, $V_A = 2.7V$, Code 80h

- Note 1:** $V_{AB} = V_{DD}$, no connection on wiper.
- 2:** Rheostat position non-linearity R-INL is the deviation from an ideal value measured between the maximum resistance and the minimum resistance wiper positions. R-DNL measures the relative step change from the ideal between successive tap positions. $I_W = V_{DD}/R$ for +3V or +5V for 50 kΩ version. See Figure 2-26 for test circuit.
- 3:** INL and DNL are measured at V_W with the device configured in the voltage divider or potentiometer mode. $V_A = V_{DD}$ and $V_B = 0V$. DNL specification limits of ± 1 LSB max are specified monotonic operating conditions. See Figure 2-25 for test circuit.
- 4:** Resistor terminals A,B and W have no restrictions on polarity with respect to each other. Full-scale and zero-scale error were measured using Figure 2-25.
- 5:** Measured at V_W pin where the voltage on the adjacent V_W pin is swinging full scale.
- 6:** Supply current is independent of current through the potentiometers.

MCP41XXX/42XXX

DC CHARACTERISTICS: 100 kΩ VERSION

Electrical Characteristics: Unless otherwise indicated, $V_{DD} = +2.7V$ to $5.5V$, $T_A = -40^\circ C$ to $+85^\circ C$ (TSSOP devices are only specified at $+25^\circ C$ and $+85^\circ C$). Typical specifications represent values for $V_{DD} = 5V$, $V_{SS} = 0V$, $V_B = 0V$, $T_A = +25^\circ C$.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Rheostat Mode						
Nominal Resistance	R	70	100	130	kΩ	$T_A = +25^\circ C$ (Note 1)
Rheostat Differential Non-Linearity	R-DNL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Integral Non-Linearity	R-INL	-1	$\pm 1/4$	+1	LSB	Note 2
Rheostat Tempco	$\Delta R_{AB}/\Delta T$	—	800	—	ppm/°C	
Wiper Resistance	R_W	—	125	175	Ω	$V_{DD} = 5.5V$, $I_W = 1$ mA, code 00h
	R_W	—	175	250	Ω	$V_{DD} = 2.7V$, $I_W = 1$ mA, code 00h
Wiper Current	I_W	-1	—	+1	mA	
Nominal Resistance Match	$\Delta R/R$	—	0.2	1	%	MCP42010 only, P0 to P1; $T_A = +25^\circ C$
Potentiometer Divider						
Resolution	N	8	—	—	Bits	
Monotonicity	N	8	—	—	Bits	
Differential Non-Linearity	DNL	-1	$\pm 1/4$	+1	LSB	Note 3
Integral Non-Linearity	INL	-1	$\pm 1/4$	+1	LSB	Note 3
Voltage Divider Tempco	$\Delta V_W/\Delta T$	—	1	—	ppm/°C	Code 80h
Full-Scale Error	V_{WFSE}	-1	-0.25	0	LSB	Code FFh, $V_{DD} = 5V$, see Figure 2-25
	V_{WFSE}	-1	-0.35	0	LSB	Code FFh, $V_{DD} = 3V$, see Figure 2-25
Zero-Scale Error	V_{WZSE}	0	+0.25	+1	LSB	Code 00h, $V_{DD} = 5V$, see Figure 2-25
	V_{WZSE}	0	+0.35	+1	LSB	Code 00h, $V_{DD} = 3V$, see Figure 2-25
Resistor Terminals						
Voltage Range	$V_{A,B,W}$	0	—	V_{DD}		Note 4
Capacitance (CA or CB)	—	—	11	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Capacitance	C_W	—	5.6	—	pF	$f = 1$ MHz, Code = 80h, see Figure 2-30
Dynamic Characteristics (All dynamic characteristics use $V_{DD} = 5V$.)						
Bandwidth -3dB	BW	—	145	—	MHz	$V_B = 0V$, Measured at Code 80h, Output Load = 30 pF
Settling Time	t_S	—	18	—	μs	$V_A = V_{DD}$, $V_B = 0V$, $\pm 1\%$ Error Band, Transition from Code 00h to Code 80h, Output Load = 30 pF
Resistor Noise Voltage	e_{NNB}	—	29	—	nV/ \sqrt{Hz}	$V_A = \text{Open}$, Code 80h, $f = 1$ kHz
Crosstalk	C_T	—	-95	—	dB	$V_A = V_{DD}$, $V_B = 0V$ (Note 5)
Digital Inputs/Outputs (CS, SCK, SI, SO) See Figure 2-12 for RS and SHDN pin operation.						
Schmitt Trigger High-Level Input Voltage	V_{IH}	$0.7V_{DD}$	—	—	V	
Schmitt Trigger Low-Level Input Voltage	V_{IL}	—	—	$0.3V_{DD}$	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	$0.05V_{DD}$	—		
Low-Level Output Voltage	V_{OL}	—	—	0.40	V	$I_{OL} = 2.1$ mA, $V_{DD} = 5V$
High-Level Output Voltage	V_{OH}	$V_{DD} - 0.5$	—	—	V	$I_{OH} = -400$ μA, $V_{DD} = 5V$
Input Leakage Current	I_{LI}	-1	—	+1	μA	$CS = V_{DD}$, $V_{IN} = V_{SS}$ or V_{DD} ; includes V_A SHDN=0
Pin Capacitance (All inputs/outputs)	C_{IN}, C_{OUT}	—	10	—	pF	$V_{DD} = 5.0V$, $T_A = +25^\circ C$, $f_c = 1$ MHz
Power Requirements						
Operating Voltage Range	V_{DD}	2.7	—	5.5	V	
Supply Current, Active	I_{DDA}	—	340	500	μA	$V_{DD} = 5.5V$, $\overline{CS} = V_{SS}$, $f_{SCK} = 10$ MHz, SO = Open, Code FFh (Note 6)
Supply Current, Static	I_{DDS}	—	0.01	1	μA	\overline{CS} , SHDN, RS = $V_{DD} = 5.5V$, SO = Open (Note 6)
Power Supply Sensitivity	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 4.5V - 5.5V$, $V_A = 4.5V$, Code 80h
	PSS	—	0.0015	0.0035	%/%	$V_{DD} = 2.7V - 3.3V$, $V_A = 2.7V$, Code 80h

- Note 1:** $V_{AB} = V_{DD}$, no connection on wiper.
2: Rheostat position non-linearity R-INL is the deviation from an ideal value measured between the maximum resistance and the minimum resistance wiper positions. R-DNL measures the relative step change from the ideal between successive tap positions. $I_W = 50$ μA for $V_{DD} = 3V$ and $I_W = 400$ μA for $V_{DD} = 5V$ for 10 kΩ version. See Figure 2-26 for test circuit.
3: INL and DNL are measured at V_W with the device configured in the voltage divider or potentiometer mode. $V_A = V_{DD}$ and $V_B = 0V$. DNL specification limits of ± 1 LSB max are specified monotonic operating conditions. See Figure 2-25 for test circuit.
4: Resistor terminals A,B and W have no restrictions on polarity with respect to each other. Full-scale and zero-scale error were measured using Figure 2-25.
5: Measured at V_W pin where the voltage on the adjacent V_W pin is swinging full-scale.
6: Supply current is independent of current through the potentiometers.

Absolute Maximum Ratings †

V _{DD}	7.0V
All inputs and outputs w.r.t. V _{SS}	-0.6V to V _{DD} +1.0V
Storage temperature	-60°C to +150°C
Ambient temp. with power applied	-60°C to +125°C
ESD protection on all pins	≥ 2 kV

† **Notice:** Stresses above those listed under "maximum ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

AC TIMING CHARACTERISTICS

Electrical Characteristics: Unless otherwise indicated, V_{DD} = +2.7V to 5.5V, T_A = -40°C to +85°C.

Parameter	Sym	Min.	Typ.	Max.	Units	Conditions
Clock Frequency	F _{CLK}	—	—	10	MHz	V _{DD} = 5V (Note 1)
Clock High Time	t _H	40	—	—	ns	
Clock Low Time	t _L	40	—	—	ns	
CS Fall to First Rising CLK Edge	t _{CSR}	40	—	—	ns	
Data Input Setup Time	t _{SU}	40	—	—	ns	
Data Input Hold Time	t _{HD}	10	—	—	ns	
SCK Fall to SO Valid Propagation Delay	t _{DO}		—	80	ns	C _L = 30 pF (Note 2)
SCK Rise to CS Rise Hold Time	t _{CHS}	30	—	—	ns	
SCK Rise to CS Fall Delay	t _{CS0}	10	—	—	ns	
CS Rise to CLK Rise Hold	t _{CS1}	100	—	—	ns	
CS High Time	t _{CSH}	40	—	—	ns	
Reset Pulse Width	t _{RS}	150	—	—	ns	Note 2
RS Rising to CS Falling Delay Time	t _{RSR}	150	—	—	ns	Note 2
CS rising to RS or SHDN falling delay time	t _{SE}	40	—	—	ns	Note 3
CS low time	t _{CSL}	100	—	—	ns	Note 3
Shutdown Pulse Width	t _{SH}	150	—	—	ns	Note 3

Note 1: When using the device in the daisy-chain configuration, maximum clock frequency is determined by a combination of propagation delay time (t_{DO}) and data input setup time (t_{SU}). Max. clock frequency is therefore ~ 5.8 MHz based on SCK rise and fall times of 5 ns, t_H = 40 ns, t_{DO} = 80 ns and t_{SU} = 40 ns.

2: Applies only to the MCP42XXX devices.

3: Applies only when using hardware pins to exit software shutdown mode, MCP42XXX only.

MCP41XXX/42XXX

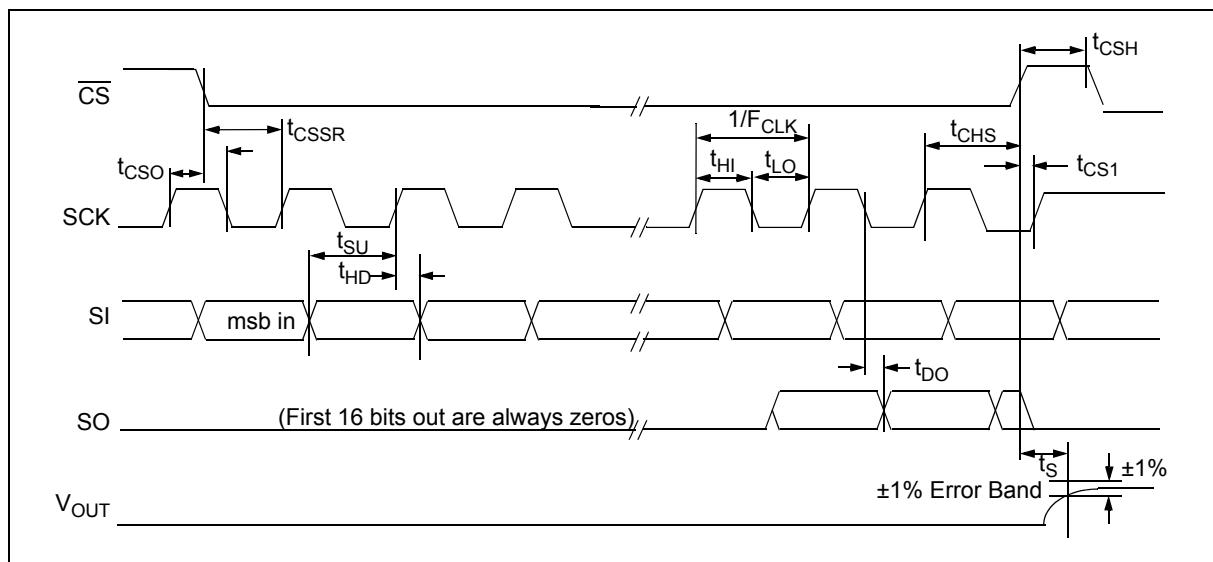


FIGURE 1-1: Detailed Serial interface Timing.

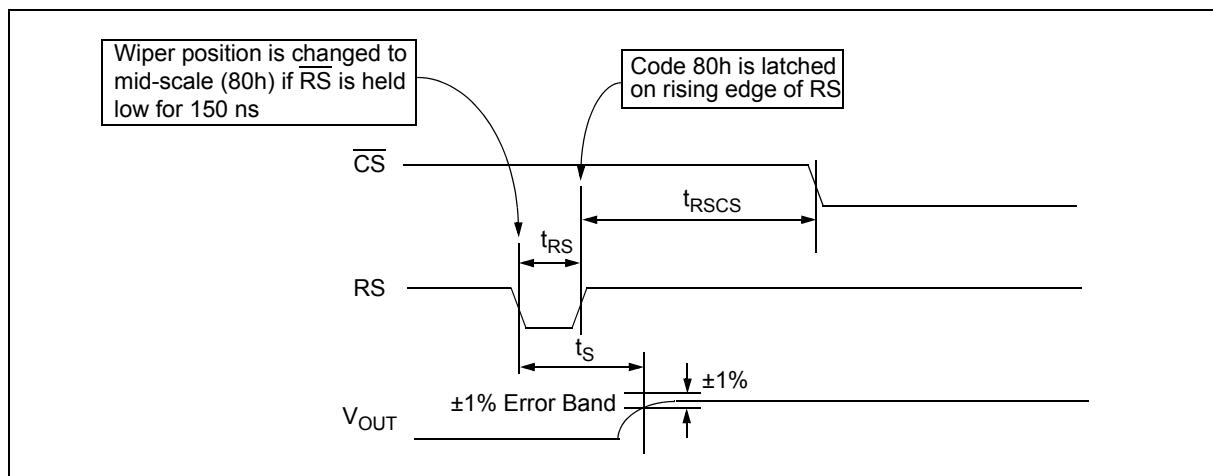


FIGURE 1-2: Reset Timing.

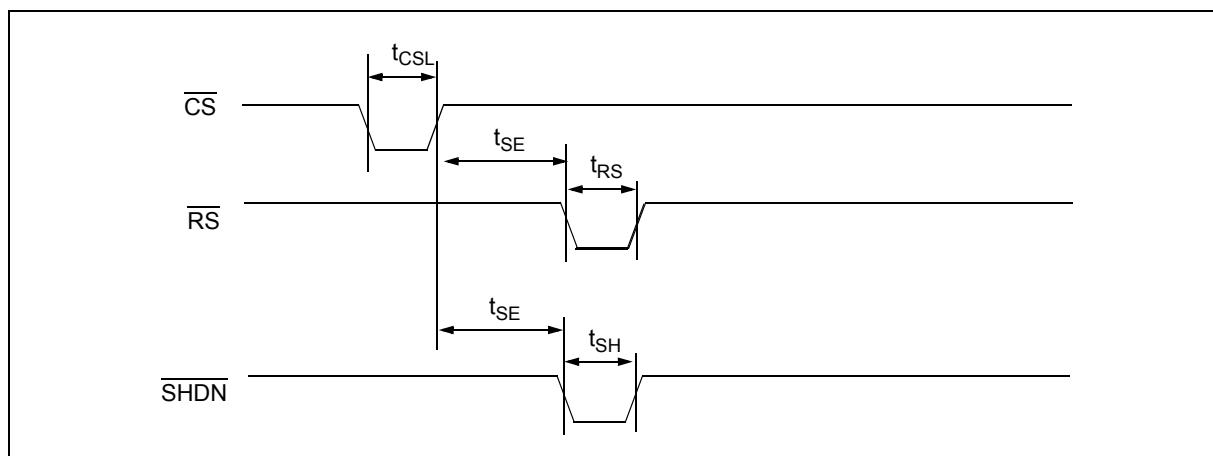


FIGURE 1-3: Software Shutdown Exit Timing.

2.0 TYPICAL PERFORMANCE CURVES

Note: The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

Note: Unless otherwise indicated, curve represents 10 k Ω , 50 k Ω and 100 k Ω devices, V_{DD} = 5V, V_{SS} = 0V, T_A = +25°C, V_B = 0V.

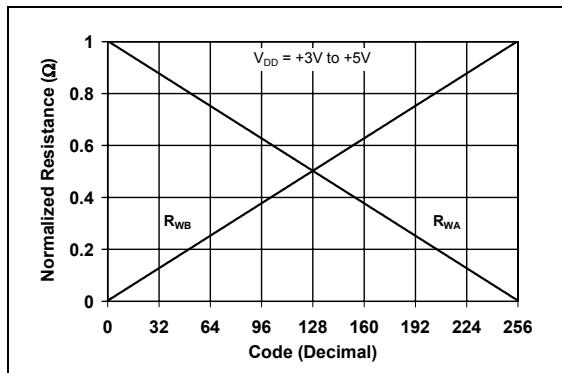


FIGURE 2-1: Normalized Wiper to End Terminal Resistance vs. Code.

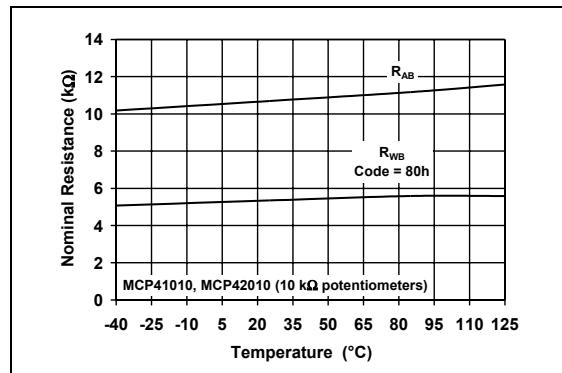


FIGURE 2-4: Nominal Resistance 10 k Ω vs. Temperature.

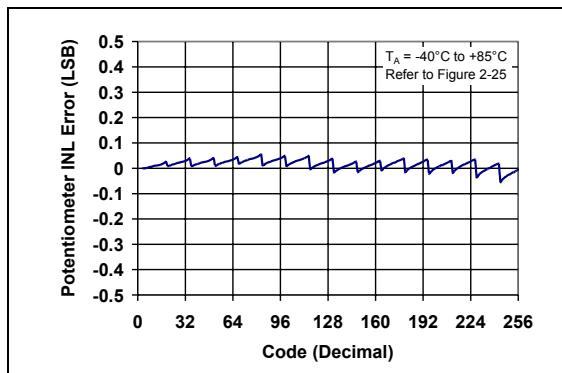


FIGURE 2-2: Potentiometer INL Error vs. Code.

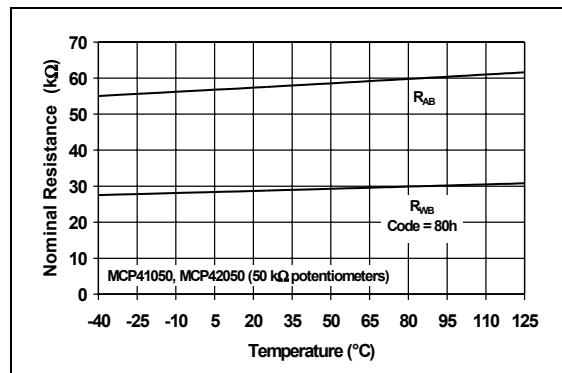


FIGURE 2-5: Nominal Resistance 50 k Ω vs. Temperature.

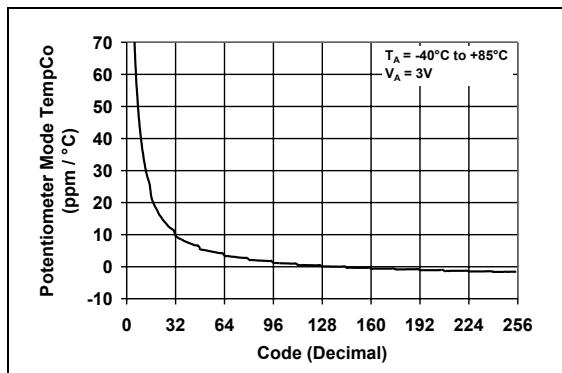


FIGURE 2-3: Potentiometer Mode Tempco vs. Code.

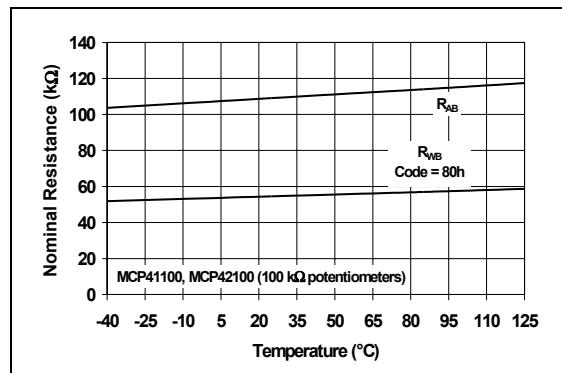


FIGURE 2-6: Nominal Resistance 100 k Ω vs. Temperature.

MCP41XXX/42XXX

Note: Unless otherwise indicated, curve represents 10 k Ω , 50 k Ω and 100 k Ω devices, V_{DD} = 5V, V_{SS} = 0V, T_A = +25°C, V_B = 0V.

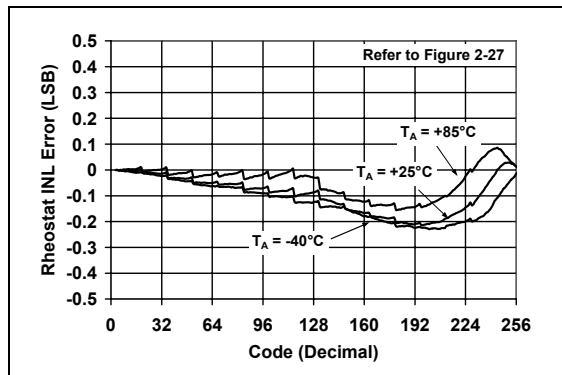


FIGURE 2-7: Rheostat INL Error vs. Code.

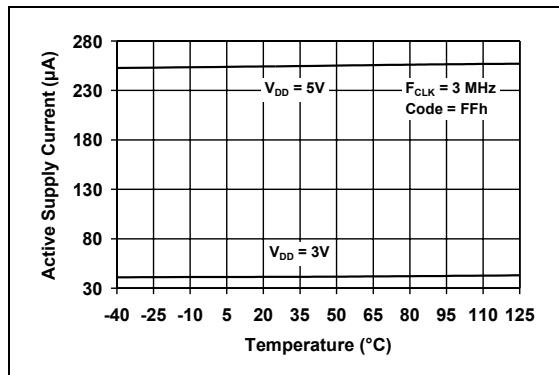


FIGURE 2-10: Active Supply Current vs. Temperature.

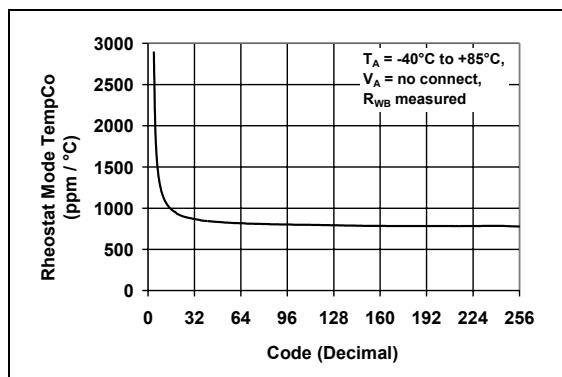


FIGURE 2-8: Rheostat Mode Tempco vs. Code.

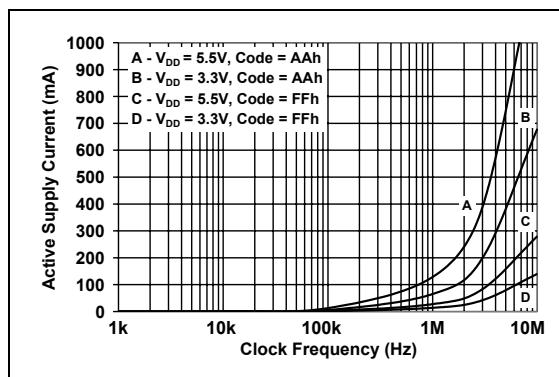


FIGURE 2-11: Active Supply Current vs. Clock Frequency.

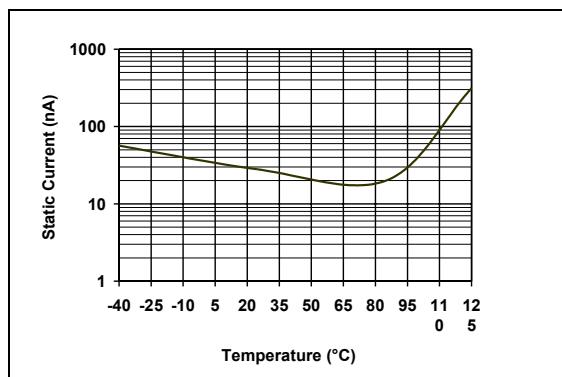


FIGURE 2-9: Static Current vs. Temperature.

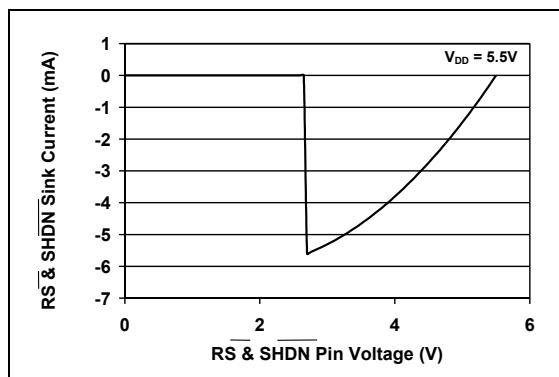


FIGURE 2-12: Reset & Shutdown Pins Current vs. Voltage.

MCP41XXX/42XXX

Note: Unless otherwise indicated, curve represents 10 k Ω , 50 k Ω and 100 k Ω devices, V_{DD} = 5V, V_{SS} = 0V, T_A = +25°C, V_B = 0V.

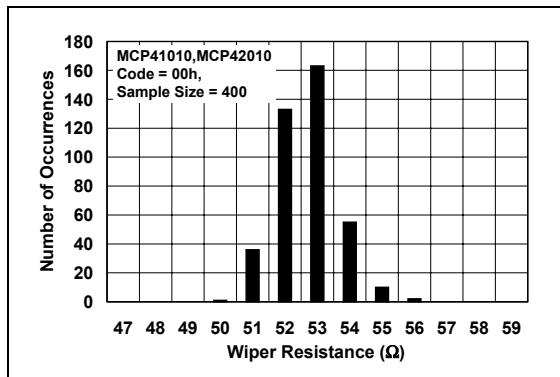


FIGURE 2-13: 10 k Ω Device Wiper Resistance Histogram.

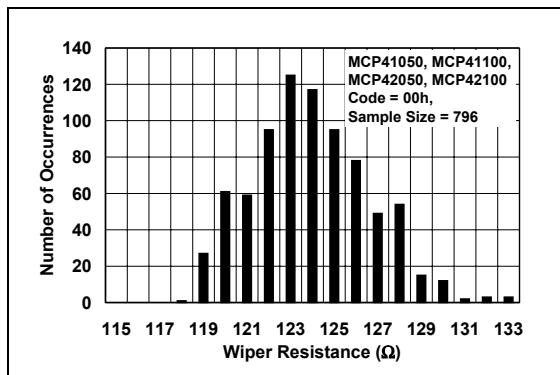


FIGURE 2-14: 50 k Ω , 100 k Ω Device Wiper Resistance Histogram.

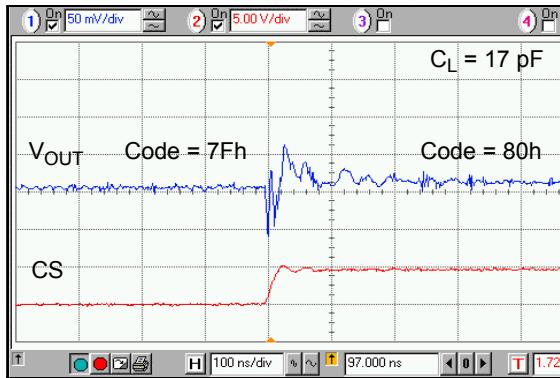


FIGURE 2-15: One Position Settling Time.

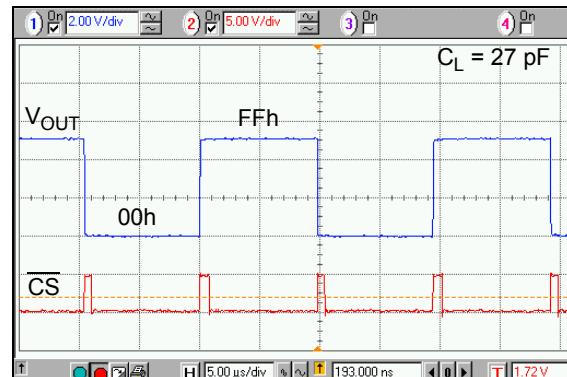


FIGURE 2-16: Full-Scale Settling Time.

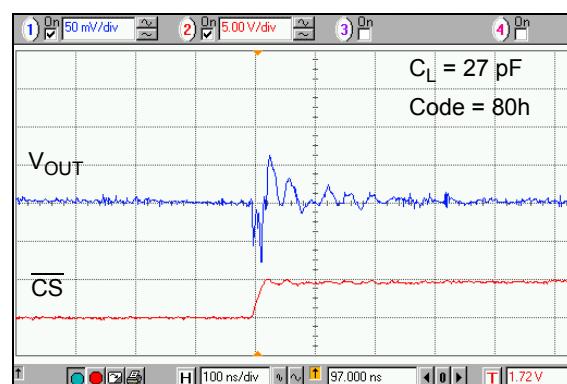


FIGURE 2-17: Digital Feed through vs. Time.

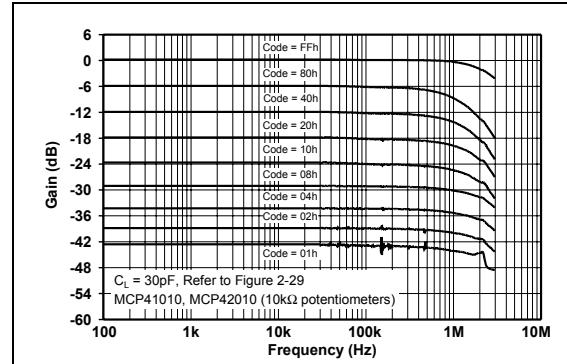


FIGURE 2-18: Gain vs. Frequency for 10 k Ω Potentiometer.

MCP41XXX/42XXX

Note: Unless otherwise indicated, curve represents 10 k Ω , 50 k Ω and 100 k Ω devices, V_{DD} = 5V, V_{SS} = 0V, T_A = +25°C, V_B = 0V.

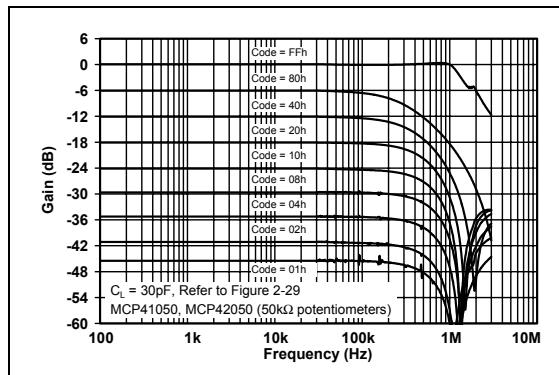


FIGURE 2-19: Gain vs. Frequency for 50k Ω Potentiometer.

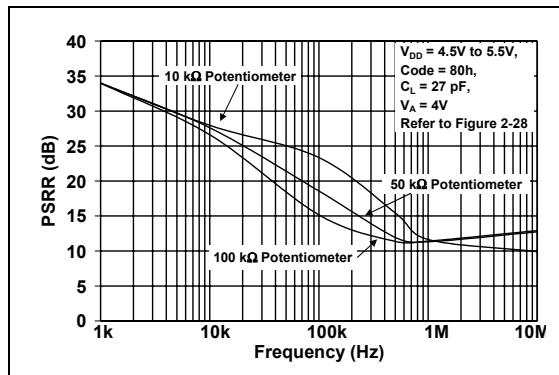


FIGURE 2-22: Power Supply Rejection Ratio vs. Frequency.

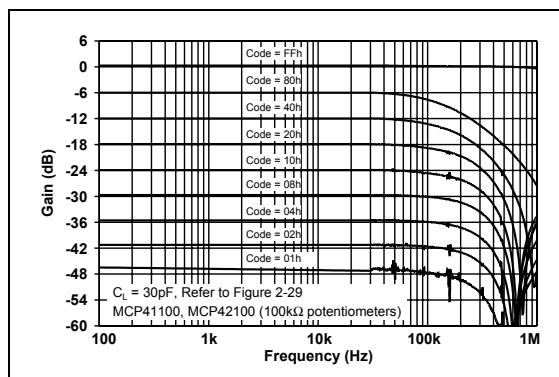


FIGURE 2-20: Gain vs. Frequency for 100k Ω Potentiometer.

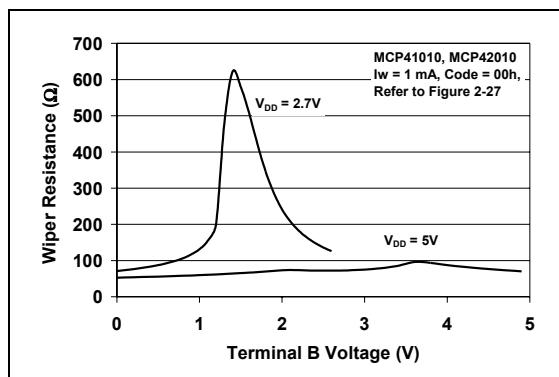


FIGURE 2-23: 10 k Ω Wiper Resistance vs. Voltage.

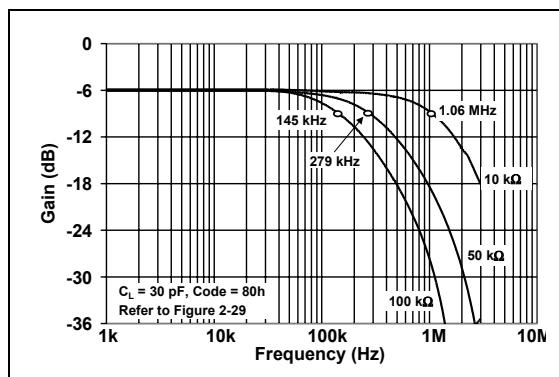


FIGURE 2-21: -3 dB Bandwidths.

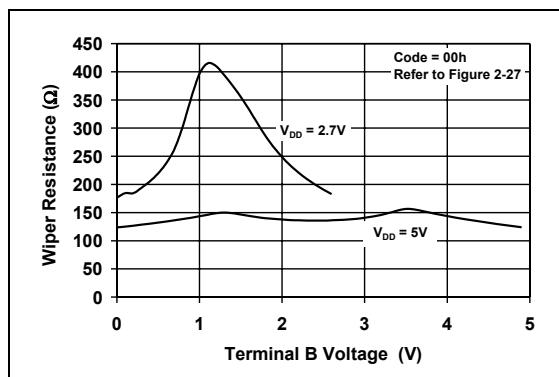


FIGURE 2-24: 50 k Ω & 100 k Ω Wiper Resistance vs. Voltage.

2.1 Parametric Test Circuits

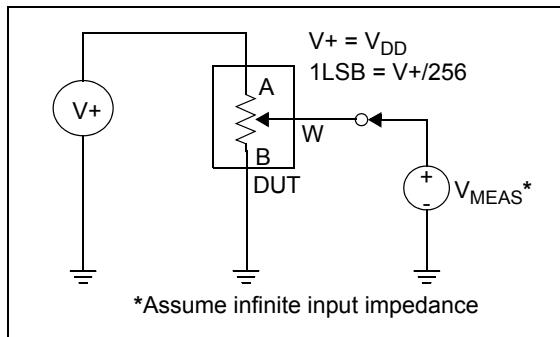


FIGURE 2-25: Potentiometer Divider Non-Linearity Error Test Circuit (DNL, INL).

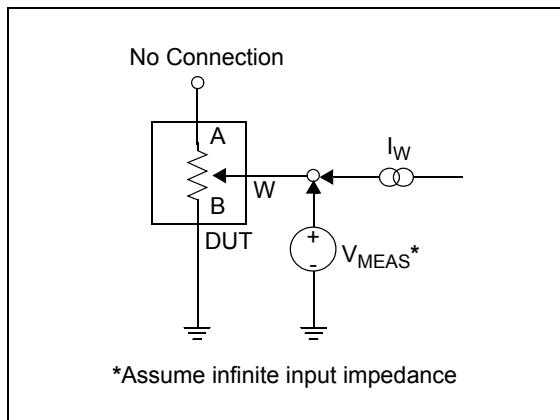


FIGURE 2-26: Resistor Position Non-Linearity Error Test Circuit (Rheostat operation DNL, INL).

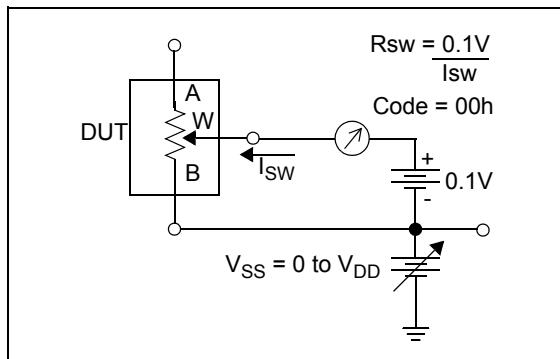


FIGURE 2-27: Wiper Resistance Test Circuit.

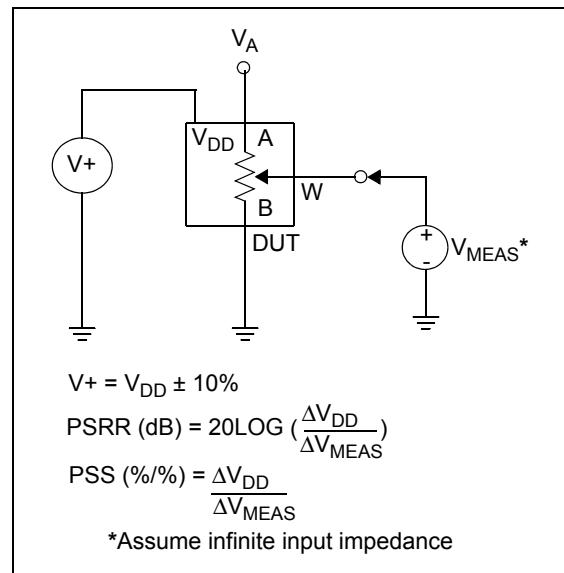


FIGURE 2-28: Power Supply Sensitivity Test Circuit (PSS, PSRR).

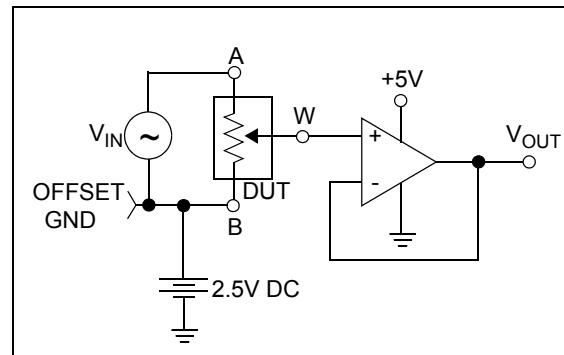


FIGURE 2-29: Gain vs. Frequency Test Circuit.

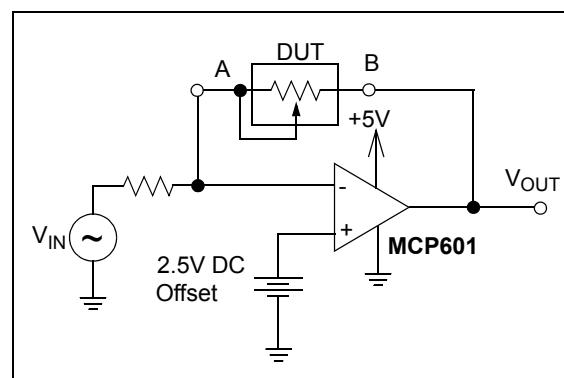


FIGURE 2-30: Capacitance Test Circuit.

MCP41XXX/42XXX

3.0 PIN DESCRIPTIONS

3.1 PA0, PA1

Potentiometer Terminal A Connection.

3.2 PB0, PB1

Potentiometer Terminal B Connection.

3.3 PW0, PW1

Potentiometer Wiper Connection.

3.4 Chip Select (CS)

This is the SPI port chip select pin and is used to execute a new command after it has been loaded into the shift register. This pin has a Schmitt Trigger input.

3.5 Serial Clock (SCK)

This is the SPI port clock pin and is used to clock-in new register data. Data is clocked into the SI pin on the rising edge of the clock and out the SO pin on the falling edge of the clock. This pin is gated to the CS pin (i.e., the device will not draw any more current if the SCK pin is toggling when the CS pin is high). This pin has a Schmitt Trigger input.

3.6 Serial Data Input (SI)

This is the SPI port serial data input pin. The command and data bytes are clocked into the shift register using this pin. This pin is gated to the CS pin (i.e., the device will not draw any more current if the SI pin is toggling when the CS pin is high). This pin has a Schmitt Trigger input.

3.7 Serial Data Output (SO) (MCP42XXX devices only)

This is the SPI port serial data output pin used for daisy-chaining more than one device. Data is clocked out of the SO pin on the falling edge of clock. This is a push-pull output and does not go to a high-impedance state when CS is high. It will drive a logic-low when CS is high.

3.8 Reset (RS) (MCP42XXX devices only)

The Reset pin will set all potentiometers to mid-scale (Code 80h) if this pin is brought low for at least 150 ns. This pin should not be toggled low when the CS pin is low. It is possible to toggle this pin when the SHDN pin is low. In order to minimize power consumption, this pin has an active pull-up circuit. The performance of this circuit is shown in Figure 2-12. This pin will draw negligible current at logic level '0' and logic level '1'. Do not leave this pin floating.

3.9 Shutdown (SHDN) (MCP42XXX devices only)

The Shutdown pin has a Schmitt Trigger input. Pulling this pin low will put the device in a power-saving mode where A terminal is opened and the B and W terminals are connected for all potentiometers. This pin should not be toggled low when the CS pin is low. In order to minimize power consumption, this pin has an active pull-up circuit. The performance of this circuit is shown in Figure 2-12. This pin will draw negligible current at logic level '0' and logic level '1'. Do not leave this pin floating.

TABLE 3-1: MCP41XXX Pins

Pin #	Name	Function
1	CS	Chip Select
2	SCK	Serial Clock
3	SI	Serial Data Input
4	V _{SS}	Ground
5	PA0	Terminal A Connection For Pot 0
6	PW0	Wiper Connection For Pot 0
7	PB0	Terminal B Connection For Pot 0
8	V _{DD}	Power

TABLE 3-2: MCP42XXX Pins

Pin #	Name	Function
1	CS	Chip Select
2	SCK	Serial Clock
3	SI	Serial Data Input
4	V _{SS}	Ground
5	PB1	Terminal B Connection For Pot 1
6	PW1	Wiper Connection For Pot 1
7	PA1	Terminal A Connection For Pot 1
8	PA0	Terminal A Connection For Pot 0
9	PW0	Wiper Connection For Pot 0
10	PB0	Terminal B Connection For Pot 0
11	RS	Reset Input
12	SHDN	Shutdown Input
13	SO	Data Out for Daisy-Chaining
14	V _{DD}	Power

4.0 APPLICATIONS INFORMATION

The MCP41XXX/42XXX devices are 256 position single and dual digital potentiometers that can be used in place of standard mechanical pots. Resistance values of 10 kΩ, 50 kΩ and 100 kΩ are available. As shown in Figure 4-1, each potentiometer is made up of a variable resistor and an 8-bit (256 position) data register that determines the wiper position. There is a nominal wiper resistance of 52Ω for the 10 kΩ version, 125Ω for the 50 kΩ and 100 kΩ versions. For the dual devices, the channel-to-channel matching variation is less than 1%. The resistance between the wiper and either of the resistor endpoints varies linearly according to the value stored in the data register. Code 00h effectively connects the wiper to the B terminal. At

power-up, all data registers will automatically be loaded with the mid-scale value (80h). The serial interface provides the means for loading data into the shift register, which is then transferred to the data registers. The serial interface also provides the means to place individual potentiometers in the shutdown mode for maximum power savings. The SHDN pin can also be used to put all potentiometers in shutdown mode and the RS pin is provided to set all potentiometers to mid-scale (80h).

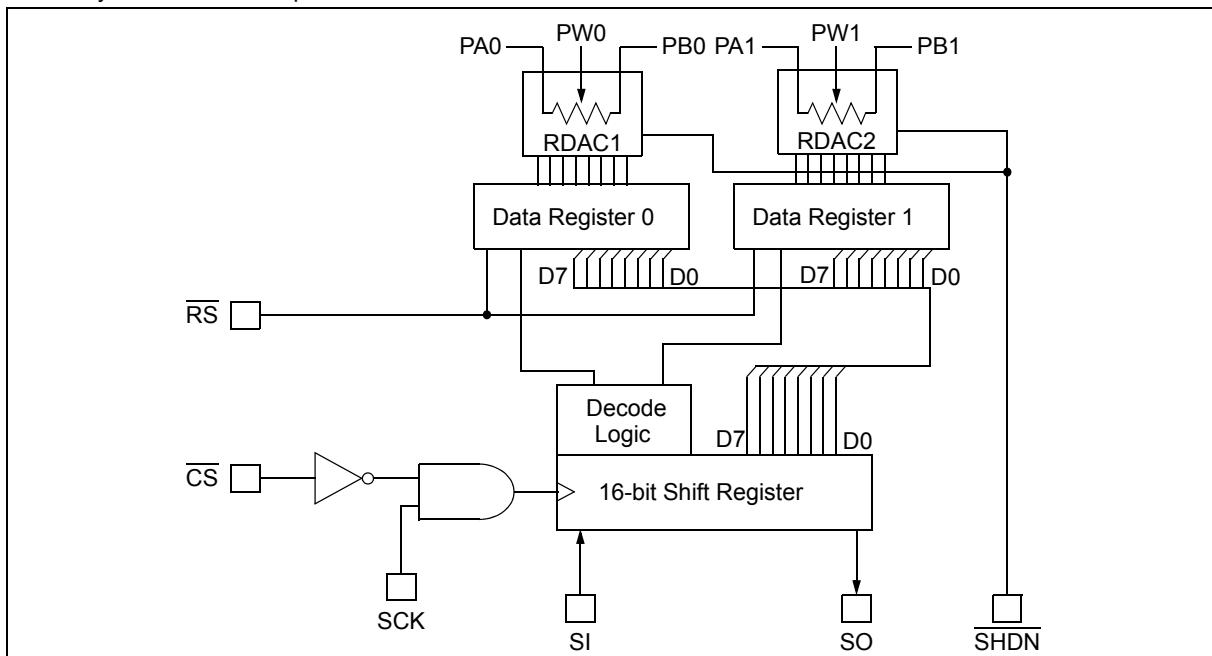
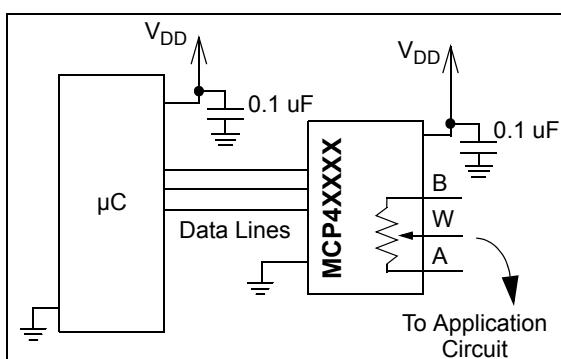


FIGURE 4-1: Block diagram showing the MCP42XXX dual digital potentiometer. Data register 0 and data register 1 are 8-bit registers allowing 256 positions for each wiper. Standard SPI pins are used with the addition of the Shutdown (SHDN) and Reset (RS) pins. As shown, reset affects the data register and wipers, bringing them to mid-scale. Shutdown disconnects the A terminal and connects the wiper to B, without changing the state of the data registers.



When laying out the circuit for your digital potentiometer, bypass capacitors should be used. These capacitors should be placed as close as possible to the device pin. A bypass capacitor value of 0.1 µF is recommended. Digital and analog traces should be separated as much as possible on the board, with no traces running underneath the device or the bypass capacitor. Extra precautions should be taken to keep traces with high-frequency signals (such as clock lines) as far as possible from analog traces. Use of an analog ground plane is recommended in order to keep the ground potential the same for all devices on the board.

MCP41XXX/42XXX

4.1 Modes of Operation

Digital potentiometer applications can be divided into two categories: rheostat mode and potentiometer, or voltage divider, mode.

4.1.1 RHEOSTAT MODE

In the rheostat mode, the potentiometer is used as a two-terminal resistive element. The unused terminal should be tied to the wiper, as shown in Figure 4-2. Note that reversing the polarity of the A and B terminals will not affect operation.

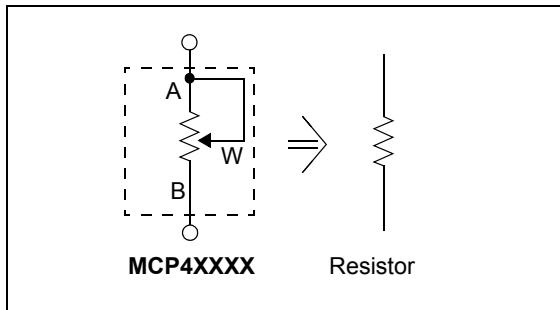


FIGURE 4-2: Two-terminal or rheostat configuration for the digital potentiometer. Acting as a resistive element in the circuit, resistance is controlled by changing the wiper setting.

Using the device in this mode allows control of the total resistance between the two nodes. The total measured resistance would be the least at code 00h, where the wiper is tied to the B terminal. The resistance at this code is equal to the wiper resistance, typically 52Ω for the 10 k Ω MCP4X010 devices, 125 Ω for the 50 k Ω (MCP4X050), and 100 k Ω (MCP4X100) devices. For the 10 k Ω device, the LSB size would be 39.0625Ω (assuming 10 k Ω total resistance). The resistance would then increase with this LSB size until the total measured resistance at code FFh would be 9985.94 Ω . The wiper will never directly connect to the A terminal of the resistor stack.

In the 00h state, the total resistance is the wiper resistance. To avoid damage to the internal wiper circuitry in this configuration, care should be taken to ensure the current flow never exceeds 1 mA.

For dual devices, the variation of channel-to-channel matching of the total resistance from A to B is less than 1%. The device-to-device matching, however, can vary up to 30%. In the rheostat mode, the resistance has a positive temperature coefficient. The change in wiper-to-end terminal resistance over temperature is shown in Figure 2-8. The most variation over temperature will occur in the first 6% of codes (code 00h to 0Fh) due to the wiper resistance coefficient affecting the total resistance. The remaining codes are dominated by the total resistance tempco R_{AB} , typically 800 ppm/ $^{\circ}\text{C}$.

4.1.2 POTENTIOMETER MODE

In the potentiometer mode, all three terminals of the device are tied to different nodes in the circuit. This allows the potentiometer to output a voltage proportional to the input voltage. This mode is sometimes called voltage divider mode. The potentiometer is used to provide a variable voltage by adjusting the wiper position between the two endpoints as shown in Figure 4-3. Note that reversing the polarity of the A and B terminals will not affect operation.

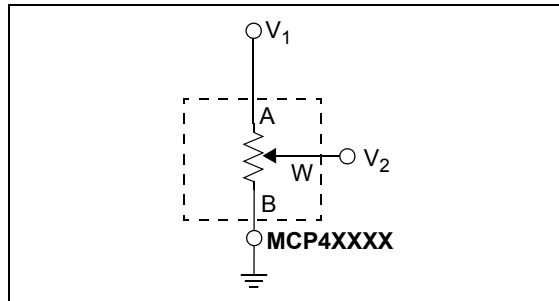


FIGURE 4-3: Three terminal or voltage divider mode.

In this configuration, the ratio of the internal resistance defines the temperature coefficient of the device. The resistor matching of the R_{WB} resistor to the R_{AB} resistor performs with a typical temperature coefficient of 1 ppm/ $^{\circ}\text{C}$ (measured at code 80h). At lower codes, the wiper resistance temperature coefficient will dominate. Figure 2-3 shows the effect of the wiper. Above the lower codes, this figure shows that 70% of the states will typically have a temperature coefficient of less than 5 ppm/ $^{\circ}\text{C}$. 30% of the states will typically have a ppm/ $^{\circ}\text{C}$ of less than 1.

4.2 Typical Applications

4.2.1 PROGRAMMABLE SINGLE-ENDED AMPLIFIERS

Potentiometers are often used to adjust system reference levels or gain. Programmable gain circuits using digital potentiometers can be realized in a number of different ways. An example of a single-supply, inverting gain amplifier is shown in Figure 4-4. Due to the high input impedance of the amplifier, the wiper resistance is not included in the transfer function. For a single-supply, non-inverting gain configuration, the circuit in Figure 4-5 can be used.

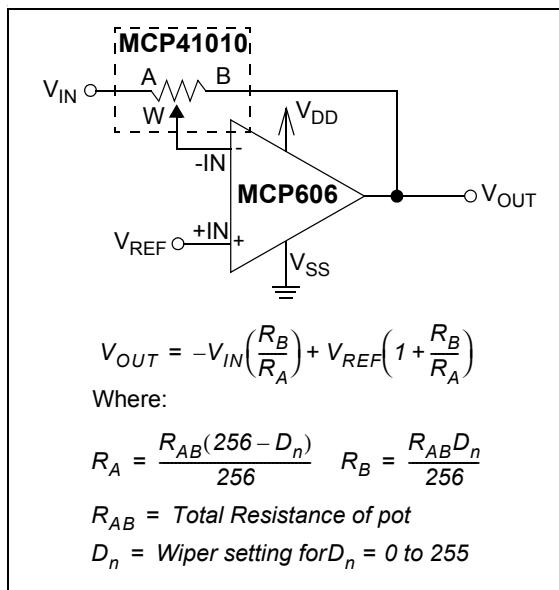


FIGURE 4-4: Single-supply, programmable, inverting gain amplifier using a digital potentiometer.

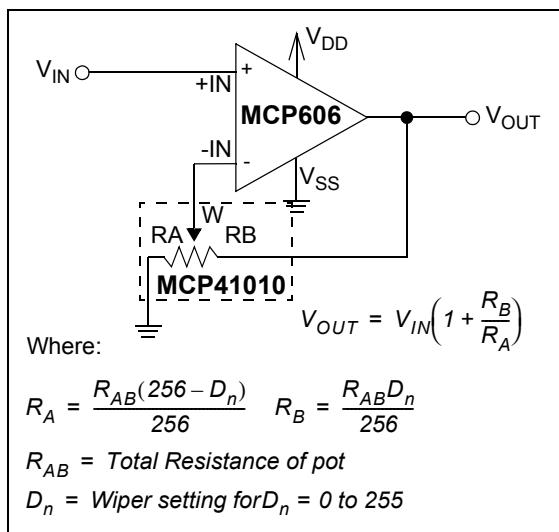


FIGURE 4-5: Single-supply, programmable, non-inverting gain amplifier.

In order for these circuits to work properly, care must be taken in a few areas. For linear operation, the analog input and output signals must be in the range of V_{SS} to V_{DD} for the potentiometer and input and output rails of the op-amp. The circuit in Figure 4-4 requires a virtual ground or reference input to the non-inverting input of the amplifier. Refer to Application Note 682, "Using Single-Supply Operational Amplifiers in Embedded Systems" (DS00682), for more details. At power-up or reset (RS), the resistance is set to mid-scale, with R_A and R_B matching. Based on the transfer function for the circuit, the gain is -1 V/V . As the code is increased and the wiper moves towards the A terminal, the gain increases. Conversely, when the wiper is moved towards the B terminal, the gain decreases. Figure 4-6 shows this relationship. Notice the pseudo-logarithmic gain around decimal code 128. As the wiper approaches either terminal, the step size in the gain calculation increases dramatically. Due to the mismatched ratio of R_A and R_B at the extreme high and low codes, small increments in wiper position can dramatically affect the gain. As shown in Figure 4-3, recommended gains lie between 0.1 and 10 V/V .

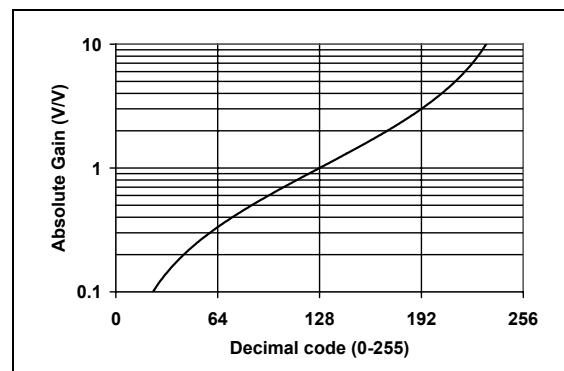


FIGURE 4-6: Gain vs. Code for inverting and differential amplifier circuits.

4.2.2 PROGRAMMABLE DIFFERENTIAL AMPLIFIER

An example of a differential input amplifier using digital potentiometers is shown in Figure 4-7. For the transfer function to hold, both pots must be programmed to the same code. The resistor-matching from channel-to-channel within a dual device can be used as an advantage in this circuit. This circuit will also show stable operation over temperature due to the low potentiometer temperature coefficient. Figure 4-6 also shows the relationship between gain and code for this circuit. As the wiper approaches either terminal, the step size in the gain calculation increases dramatically. This circuit is recommended for gains between 0.1 and 10 V/V .

MCP41XXX/42XXX

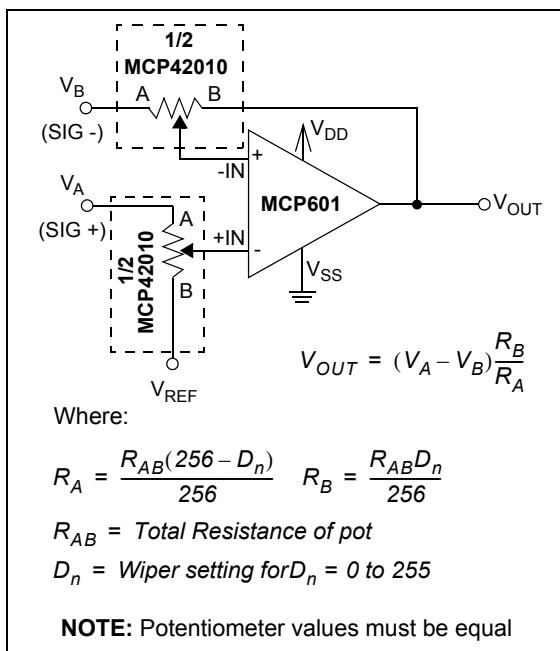


FIGURE 4-7: Single Supply programmable differential amplifier using digital potentiometers.

4.2.3 PROGRAMMABLE OFFSET TRIM

For applications requiring only a programmable voltage reference, the circuit in Figure 4-8 can be used. This circuit shows the device used in the potentiometer mode along with two resistors and a buffered output. This creates a circuit with a linear relationship between voltage-out and programmed code. Resistors R₁ and R₂ can be used to increase or decrease the output voltage step size. The potentiometer in this mode is stable over temperature. The operation of this circuit over temperature is shown in Figure 2-3. The worst performance over temperature will occur at the lower codes due to the dominating wiper resistance. R₁ and R₂ can also be used to affect the boundary voltages, thereby eliminating the use of these lower codes.

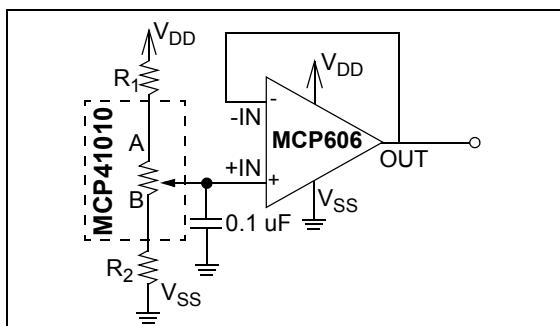


FIGURE 4-8: By changing the values of R₁ and R₂, the voltage output resolution of this programmable voltage reference circuit is affected.

4.3 Calculating Resistances

When programming the digital potentiometer settings, the following equations can be used to calculate the resistances. Programming code 00h effectively brings the wiper to the B terminal, leaving only the wiper resistance. Programming higher codes will bring the wiper closer to the A terminal of the potentiometer. The equations in Figure 4-9 can be used to calculate the terminal resistances. Figure 4-10 shows an example calculation using a 10 kΩ potentiometer.

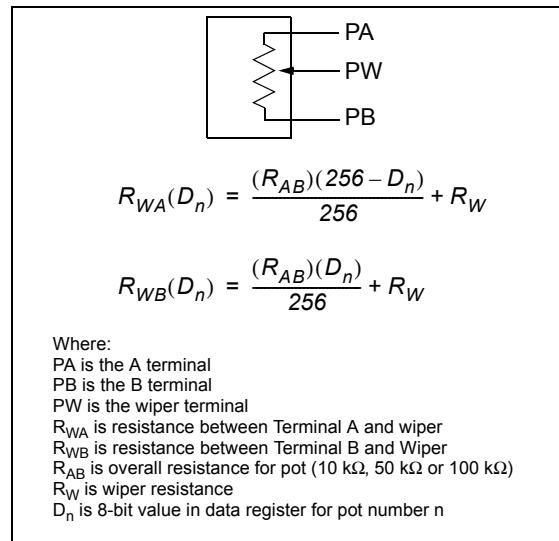


FIGURE 4-9: Potentiometer resistances are a function of code. It should be noted that, when using these equations for most feedback amplifier circuits (see Figure 4-4 and Figure 4-5), the wiper resistance can be omitted due to the high impedance input of the amplifier.

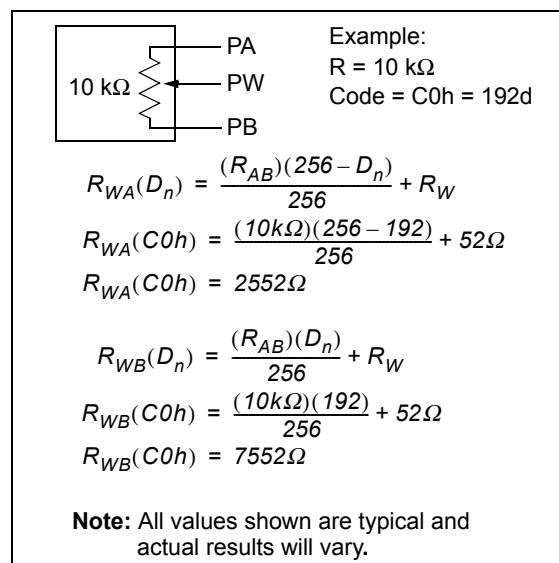


FIGURE 4-10: Example Resistance calculations.

5.0 SERIAL INTERFACE

Communications from the controller to the MCP41XXX/42XXX digital potentiometers is accomplished using the SPI serial interface. This interface allows three commands:

1. Write a new value to the potentiometer data register(s).
2. Cause a channel to enter low power shutdown mode.
3. NOP (No Operation) command.

Executing any command is accomplished by setting CS low and then clocking-in a command byte followed by a data byte into the 16-bit shift register. The command is executed when CS is raised. Data is clocked-in on the rising edge of clock and out the SO pin on the falling edge of the clock (see Figure 5-1). The device will track the number of clocks (rising edges) while CS is low and will abort all commands if the number of clocks is not a multiple of 16.

5.1 Command Byte

The first byte sent is always the command byte, followed by the data byte. The command byte contains two command select bits and two potentiometer select bits. Unused bits are ‘don’t care’ bits. The command select bits are summarized in Figure 5-2. The command select bits C1 and C0 (bits 4:5) of the command byte determine which command will be executed. If the command bits are both 0’s or 1’s, then a NOP command will be executed once all 16 bits have been loaded. This command is useful when using the daisy-chain configuration. When the command bits are 0,1, a write command will be executed with the 8 bits sent in the data byte. The data will be written to the potentiometer(s) determined by the potentiometer select bits. If the command bits are 1,0, then a shutdown command will be executed on the potentiometers determined by the potentiometer select bits.

For the MCP42XXX devices, the potentiometer select bits P1 and P0 (bits 0:1) determine which potentiometers are to be acted upon by the command. A corresponding ‘1’ in the position signifies that the command for that potentiometer will get executed, while a ‘0’ signifies that the command will not effect that potentiometer (see Figure 5-2).

5.2 Writing Data Into Data Registers

When new data is written into one or more of the potentiometer data registers, the write command is followed by the data byte for the new value. The command select bits C1, C0 are set to 0,1. The potentiometer selection bits P1 and P0 allow new values to be written to potentiometer 0, potentiometer 1 (or both) with a single command. A ‘1’ for either P1 or P0 will cause the data to be written to the respective data register and a ‘0’ for P1 or P0 will cause no change. See Figure 5-2 for the command format summary.

5.3 Using The Shutdown Command

The shutdown command allows the user to put the application circuit into a power-saving mode. In this mode, the A terminal is open-circuited and the B and W terminals are shorted together. The command select bits C1, C0 are set to 1,0. The potentiometer selection bits P1 and P0 allow each potentiometer to be shutdown independently. If either P1 or P0 are high, the respective potentiometer will enter shutdown mode. A ‘0’ for P1 or P0 will have no effect. The eight data bits following the command byte still need to be transmitted for the shutdown command, but they are ‘don’t care’ bits. See Figure 5-2 for command format summary. Once a particular potentiometer has entered the shutdown mode, it will remain in this mode until:

- A new value is written to the potentiometer data register, provided that the SHDN pin is high. The device will remain in the shutdown mode until the rising edge of the CS is detected, at which time the device will come out of shutdown mode and the new value will be written to the data register(s). If the SHDN pin is low when the new value is received, the registers will still be set to the new value, but the device will remain in shutdown mode. This scenario assumes that a valid command was received. If an invalid command was received, the command will be ignored and the device will remain in the shutdown mode.

It is also possible to use the hardware shutdown pin and reset pin to remove a device from software shutdown. To do this, a low pulse on the chip select line must first be sent. For multiple devices, sharing a single SHDN or RESET line allows you to pick an individual device on that chain to remove from software shutdown mode. See Figure 1-3 for timing. With a preceding chip select pulse, either of these situations will also remove a device from software shutdown:

- A falling edge is seen on the RS pin and held low for at least 150 ns, provided that the SHDN pin is high. If the SHDN pin is low, the registers will still be set to mid-scale, but the device will remain in shutdown mode. This condition assumes that CS is high, as bringing the RS pin low while CS is low is an invalid state and results are indeterminate.
- A rising edge on the SHDN pin is seen after being low for at least 100 ns, provided that the CS pin is high. Toggling the SHDN pin low while CS is low is an invalid state and results are indeterminate.
- The device is powered-down and back up.

Note: The hardware SHDN pin will always put the device in shutdown regardless of whether a potentiometer has already been put in the shutdown mode using the software command.

MCP41XXX/42XXX

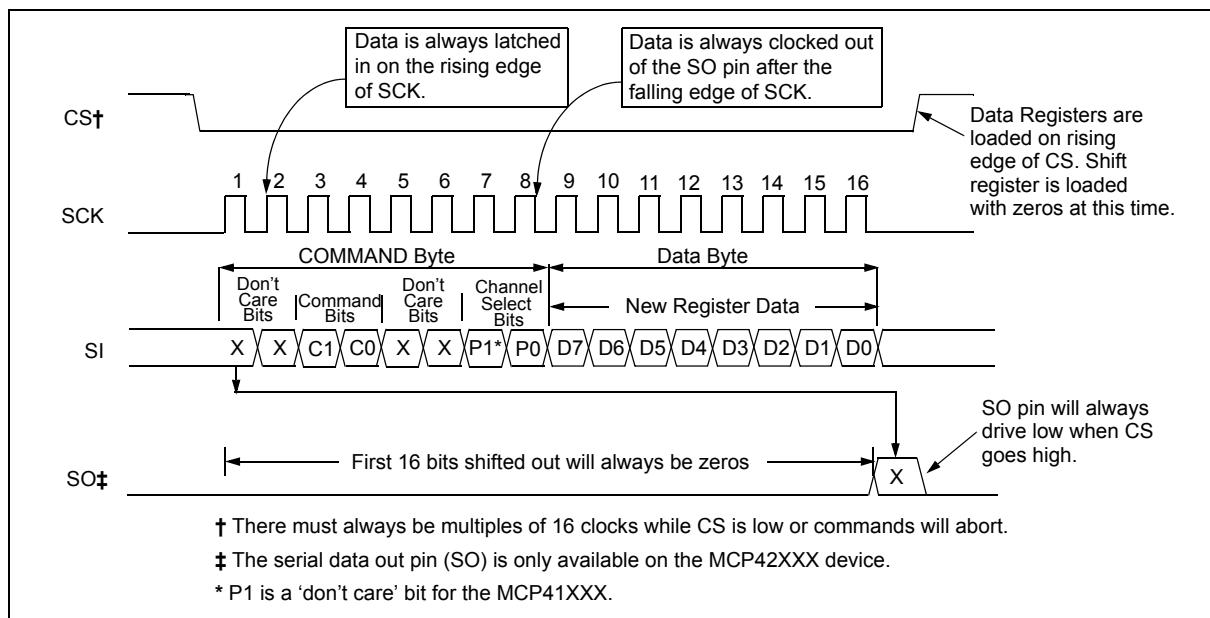


FIGURE 5-1: Timing Diagram for Writing Instructions or Data to a Digital Potentiometer.

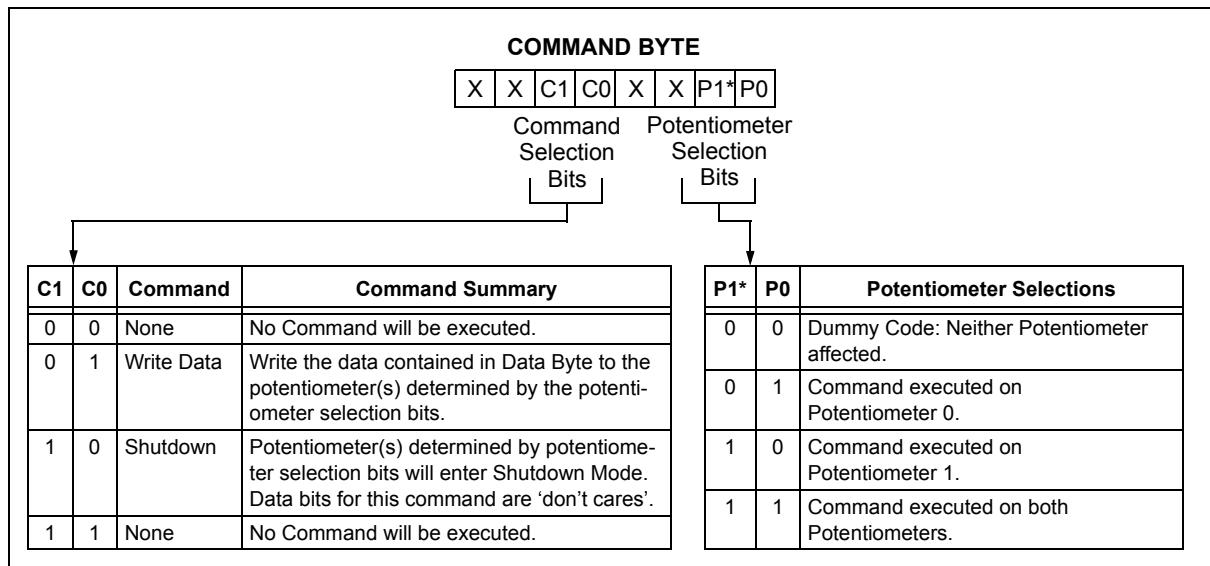


FIGURE 5-2: Command Byte Format.

5.4 Daisy-Chain Configuration

Multiple MCP42XXX devices can be connected in a daisy-chain configuration, as shown in Figure 5-4, by connecting the SO pin from one device to the SI pin on the next device. The data on the SO pin is the output of the 16-bit shift register. The daisy-chain configuration allows the system designer to communicate with several devices without using a separate CS line for each device. The example shows a daisy-chain configuration with three devices, although any number of devices (with or without the same resistor values) can be configured this way. While it is not possible to use a MCP41XXX at the beginning or middle of a daisy-chain (because it does not provide the serial data out (SO) pin), it is possible to use the device at the end of a chain. As shown in the timing diagram in Figure 5-3, data will be clocked-out of the SO pin on the falling edge of the clock. The SO pin has a CMOS push-pull output and will drive low when CS goes high. SO will not go to a high-impedance state when CS is held high.

When using the daisy-chain configuration, the maximum clock speed possible is reduced to ~5.8 MHz, because of the propagation delay of the data coming out of the SO pin.

When using the daisy-chain configuration, keep in mind that the shift register of each device is automatically loaded with zeros whenever a command is executed (CS = high). Because of this, the first 16 bits that come out of the SO pin once the CS line goes low will always be zeros. This means that when the first command is being loaded into a device, it will always shift a NOP command into the next device on the chain because the command bits (and all the other bits) will be zeros. This feature makes it necessary only to send command and data bytes to the device farthest down the chain that needs a new command. For example, if there were three devices on the chain and it was desired to send a command to the device in the middle, only 32 bytes of data need to be transmitted. The last device on the chain will have a NOP loaded from the previous device so no registers will be affected when the CS pin is raised to execute the command. **The user must always ensure that multiples of 16 clocks are always provided (while CS is low), as all commands will abort if the number of clocks provided is not a multiple of 16.**

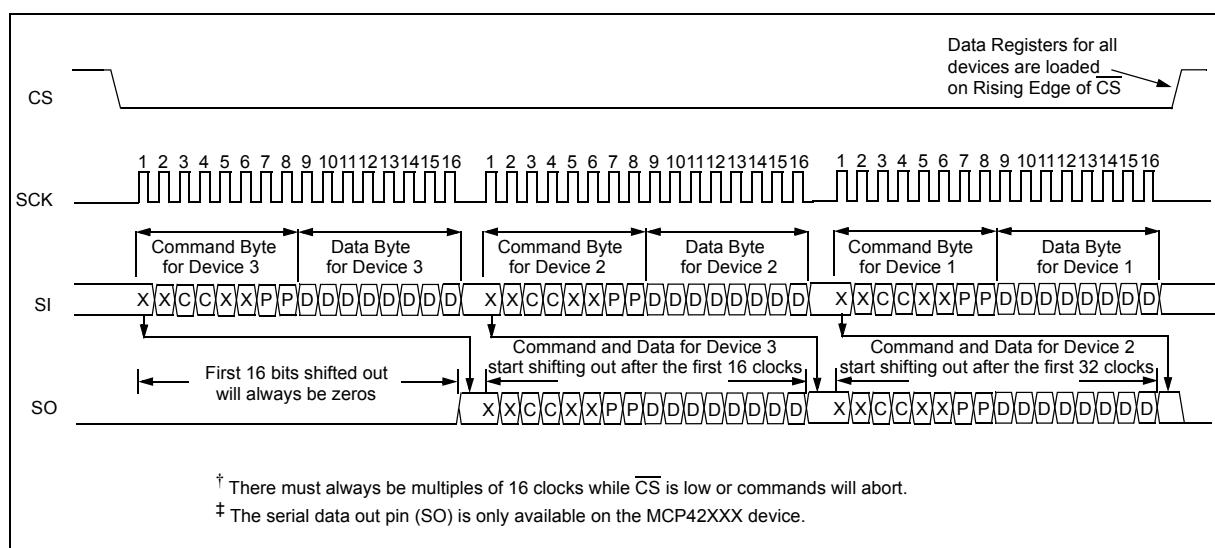


FIGURE 5-3: Timing Diagram for Daisy-Chain Configuration.

MCP41XXX/42XXX

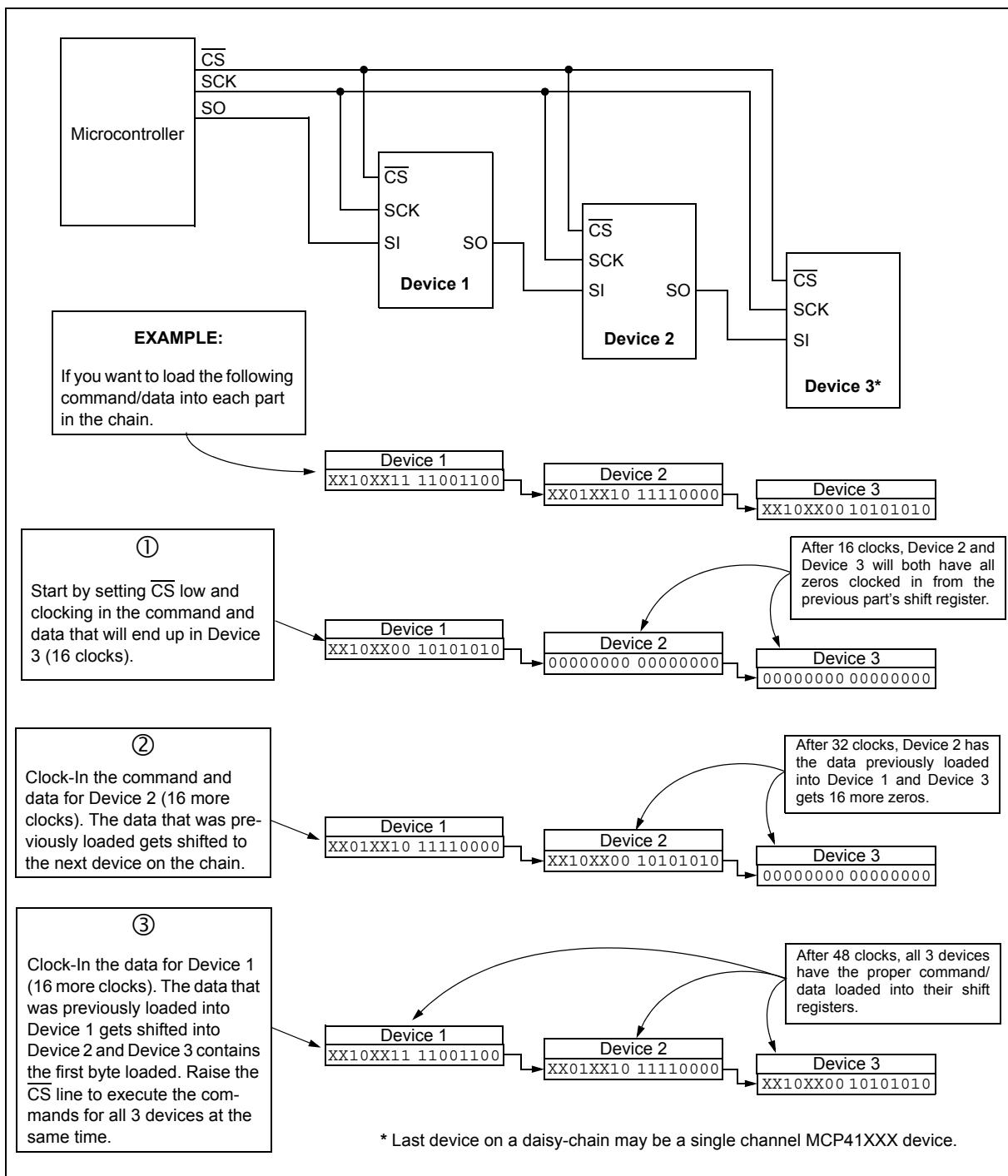


FIGURE 5-4: Daisy-Chain Configuration.

5.5 Reset (RS) Pin Operation

The Reset pin (RS) will automatically set all potentiometer data latches to mid-scale (Code 80h) when pulled low (provided that the pin is held low at least 150 ns and CS is high). The reset will execute regardless of the position of the SCK, SHDN and SI pins. It is possible to toggle RS low and back high while SHDN is low. In this case, the potentiometer registers will reset to mid-scale, but the potentiometer will remain in shutdown mode until the SHDN pin is raised.

Note: Bringing the RS pin low while the CS pin is low constitutes an invalid operating state and will result in indeterminate results when RS and/or CS are brought high.

5.6 Shutdown (SHDN) Pin Operation

When held low, the shutdown pin causes the application circuit to go into a power-saving mode by open-circuiting the A terminal and shorting the B and W terminals for all potentiometers. Data register contents are not affected by entering shutdown mode (i.e., when the SHDN pin is raised, the data register contents are the same as before the shutdown mode was entered).

While in shutdown mode, it is still possible to clock in new values for the data registers, as well as toggling the RS pin to cause all data registers to go to mid-scale. The new values will take affect when the SHDN pin is raised.

If the device is powered-up with the SHDN pin held low, it will power-up in the shutdown mode with the data registers set to mid-scale.

Note: Bringing the SHDN pin low while the CS pin is low constitutes an invalid operating state and will result in indeterminate results when SHDN and/or CS are brought high.

5.7 Power-up Considerations

When the device is powered on, the data registers will be set to mid-scale (80h). A power-on reset circuit is utilized to ensure that the device powers up in this known state.

TABLE 5-1: TRUTH TABLE FOR LOGIC INPUTS

SCK	<u>CS</u>	<u>RS</u>	<u>SHDN</u>	Action
X	Ø	H	H	Communication is initiated with device. Device comes out of standby mode.
L	L	H	H	No action. Device is waiting for data to be clocked into shift register or CS to go high to execute command.
!	L	H	X	Shift one bit into shift register. The shift register can be loaded while the <u>SHDN</u> pin is low.
Ø	L	H	X	Shift one bit out of shift register on the SO pin. The SO pin is active while the <u>SHDN</u> pin is low.
X	!	H	H	Based on command bits, either load data from shift register into data latches or execute shutdown command. Neither command executed unless multiples of 16 clocks have been entered while <u>CS</u> is low. SO pin goes to a logic low.
X	H	H	H	Static Operation.
X	H	Ø	H	All data registers set and latched to code 80h.
X	H	Ø	L	All data registers set and latched to code 80h. Device is in hardware shutdown mode and will remain in this mode.
X	H	H	Ø	All potentiometers put into hardware shutdown mode; terminal A is open and W is shorted to B.
X	H	H	!	All potentiometers exit hardware shutdown mode. Potentiometers will also exit software shutdown mode if this rising edge occurs after a low pulse on CS. Contents of data latches are restored.

MCP41XXX/42XXX

5.8 Using the MCP41XXX/42XXX in SPI Mode 1,1

It is possible to operate the devices in SPI modes 0,0 and 1,1. The only difference between these two modes is that, when using mode 1,1, the clock idles in the high state, while in mode 0,0, the clock idles in the low state. In both modes, data is clocked into the devices on the rising edge of SCK and data is clocked out the SO pin once the falling edge of SCK. Operations using mode 0,0 are shown in Figure 5-1. The example in Figure 5-5 shows mode 1,1.

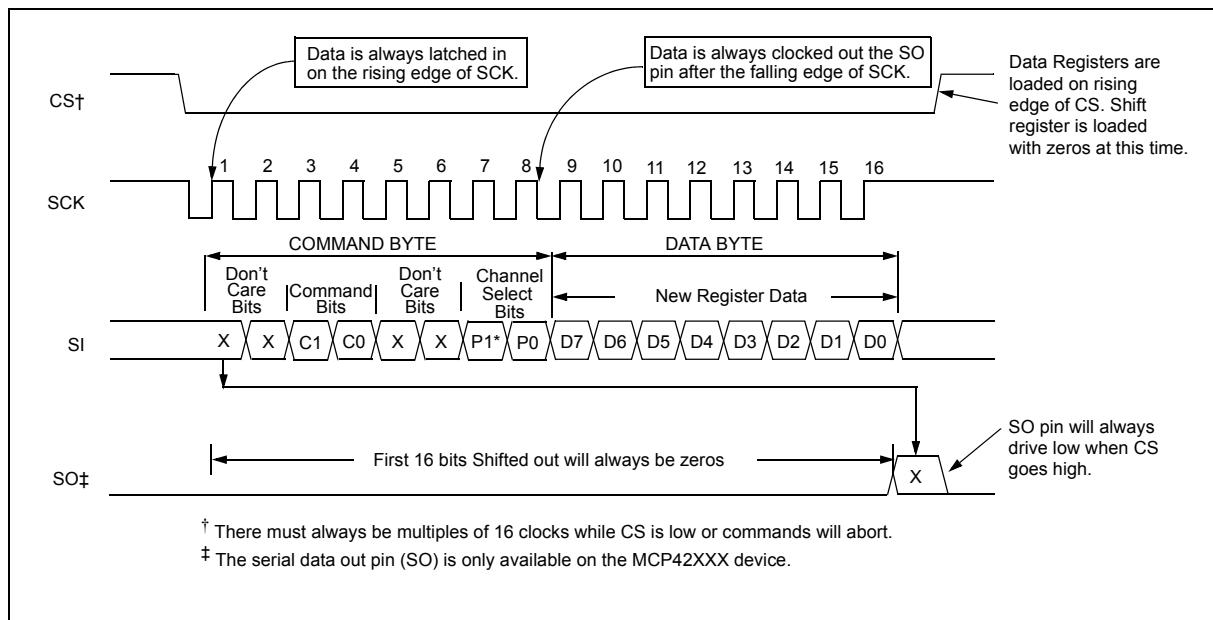


FIGURE 5-5: Timing Diagram for SPI Mode 1,1 Operation.



MCP23017/MCP23S17

16-Bit I/O Expander with Serial Interface

Features

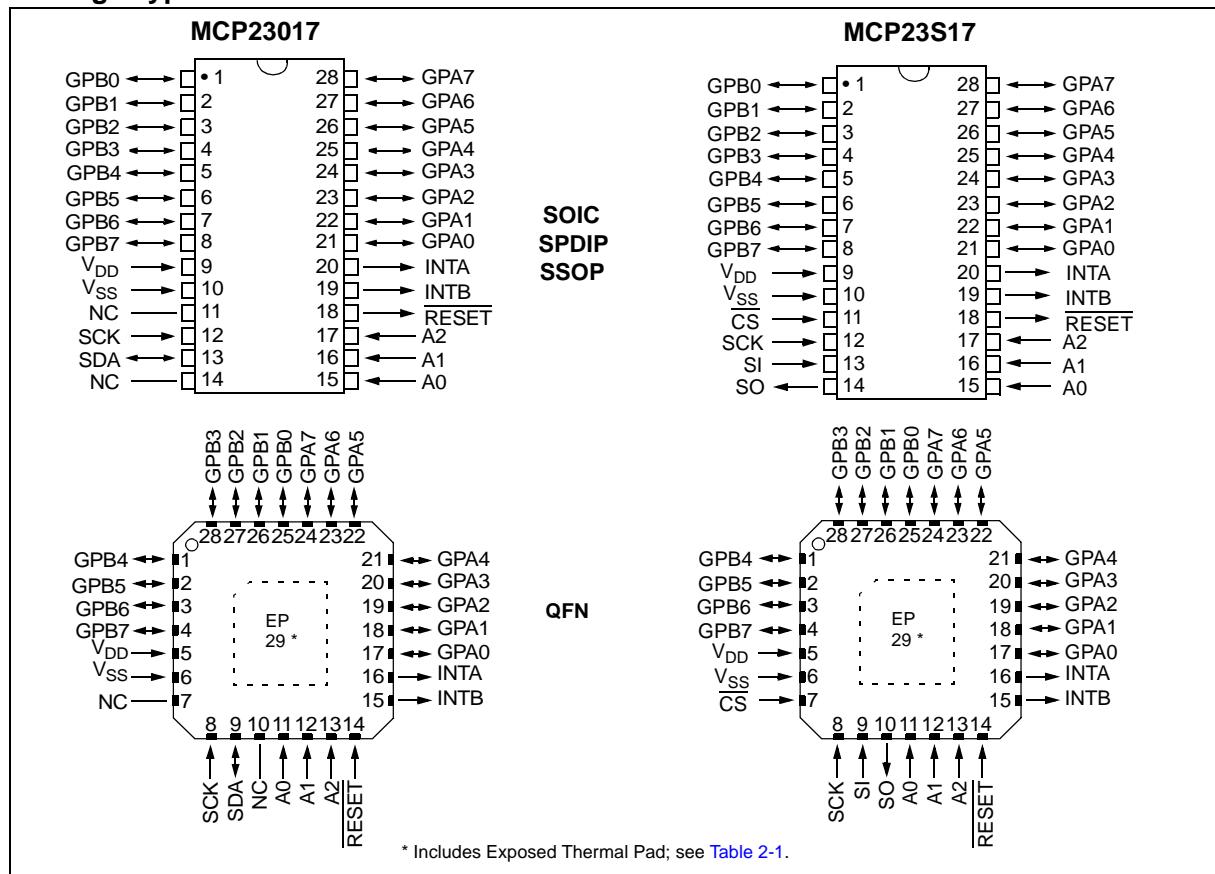
- 16-Bit Remote Bidirectional I/O Port:
 - I/O pins default to input
- High-Speed I²C Interface (**MCP23017**):
 - 100 kHz
 - 400 kHz
 - 1.7 MHz
- High-Speed SPI Interface (**MCP23S17**):
 - 10 MHz (maximum)
- Three Hardware Address Pins to Allow Up to Eight Devices On the Bus
- Configurable Interrupt Output Pins:
 - Configurable as active-high, active-low or open-drain
- INTA and INTB Can Be Configured to Operate Independently or Together

- Configurable Interrupt Source:
 - Interrupt-on-change from configured register defaults or pin changes
- Polarity Inversion Register to Configure the Polarity of the Input Port Data
- External Reset Input
- Low Standby Current: 1 µA (max.)
- Operating Voltage:
 - 1.8V to 5.5V @ -40°C to +85°C
 - 2.7V to 5.5V @ -40°C to +85°C
 - 4.5V to 5.5V @ -40°C to +125°C

Packages

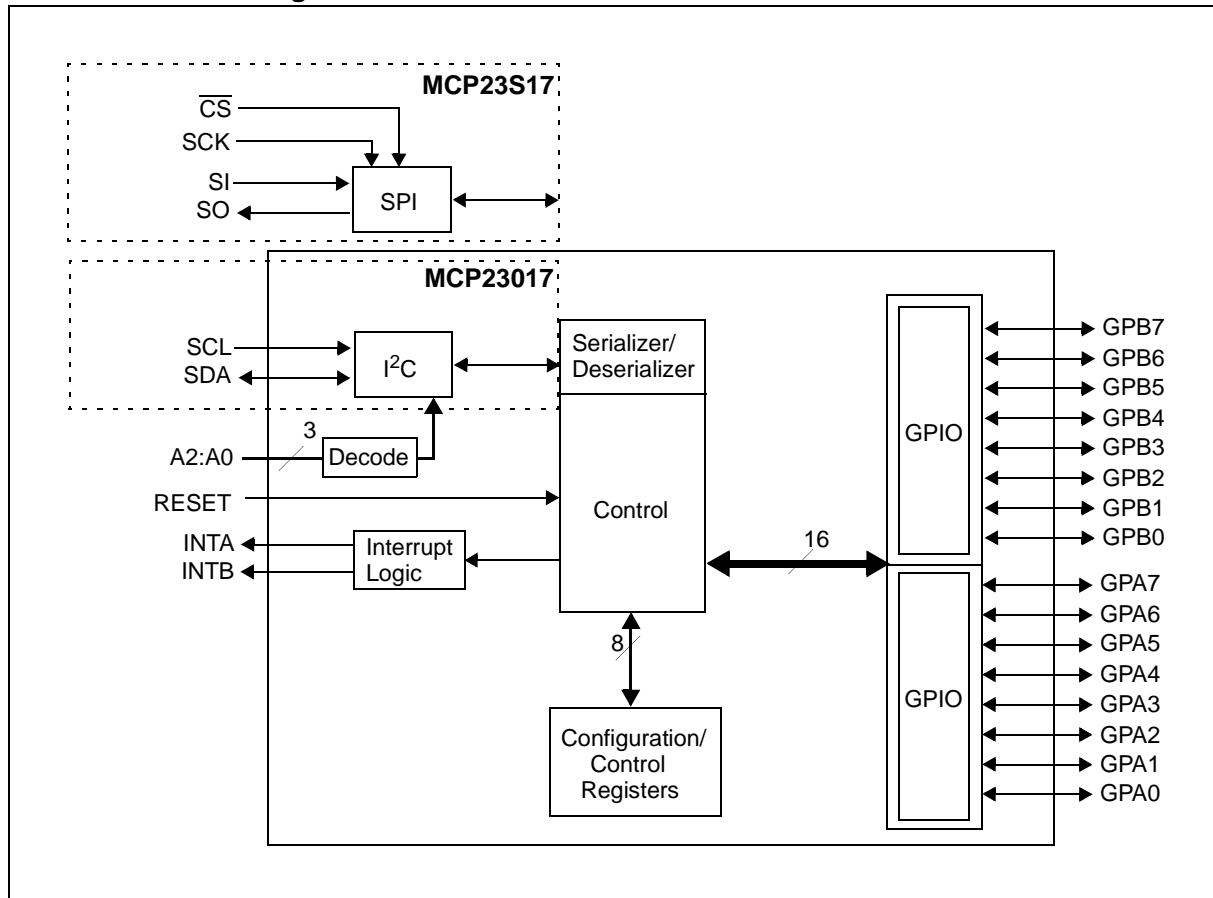
- 28-pin QFN, 6 x 6 mm Body
- 28-pin SOIC, Wide, 7.50 mm Body
- 28-pin SPDIP, 300 mil Body
- 28-pin SSOP, 5.30 mm Body

Package Types



MCP23017/MCP23S17

Functional Block Diagram



MCP23017/MCP23S17

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

Ambient temperature under bias.....	-40°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on V_{DD} with respect to V_{SS}	-0.3V to +5.5V
Voltage on all other pins with respect to V_{SS} (except V_{DD}).....	-0.6V to (V_{DD} + 0.6V)
Total power dissipation.....	700 mW
Maximum current out of V_{SS} pin	150 mA
Maximum current into V_{DD} pin	125 mA
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{DD}$).....	±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{DD}$).....	±20 mA
Maximum output current sunk by any output pin	25 mA
Maximum output current sourced by any output pin	25 mA
ESD protection on all pins (HBM:MM)4 kV:400V

† Notice: Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

MCP23017/MCP23S17

1.1 DC Characteristics

TABLE 1-1: DC CHARACTERISTICS

Electrical Specifications: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$							
Param. No.	Characteristic	Sym.	Min.	Typ. ⁽¹⁾	Max.	Units	Conditions
D001	Supply Voltage	V_{DD}	1.8	—	5.5	V	
D002	V_{DD} Start Voltage to ensure Power-on Reset	V_{POR}	—	V_{SS}	—	V	
D003	V_{DD} Rise Rate to ensure Power-on Reset	SV_{DD}	0.05	—	—	V/ms	Design guidance only. Not tested.
D004	Supply Current	I_{DD}	—	—	1	mA	$SCL/SCK = 1\text{ MHz}$
D005	Standby current	I_{DDS8}	—	—	1	μA	$-40^{\circ}C \leq T_A \leq +85^{\circ}C$
			—	—	3	μA	$4.5V \leq V_{DD} \leq 5.5V$ $+85^{\circ}C \leq T_A \leq +125^{\circ}C$ (Note 1)
Input Low Voltage							
D030	A0, A1, A2 (TTL buffer)	V_{IL}	V_{SS}	—	$0.15 V_{DD}$	V	
D031	CS, GPIO, SCL/SCK, SDA, RESET (Schmitt Trigger)	V_{IL}	V_{SS}	—	$0.2 V_{DD}$	V	
Input High Voltage							
D040	A0, A1, A2 (TTL buffer)	V_{IH}	$0.25 V_{DD} + 0.8$	—	V_{DD}	V	
D041	CS, GPIO, SCL/SCK, SDA, RESET (Schmitt Trigger)	V_{IH}	$0.8 V_{DD}$	—	V_{DD}	V	For entire V_{DD} range
Input Leakage Current							
D060	I/O port pins	I_{IL}	—	—	± 1	μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$
Output Leakage Current							
D065	I/O port pins	I_{LO}	—	—	± 1	μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$
D070	GPIO weak pull-up current	I_{PU}	40	75	115	μA	$V_{DD} = 5V$ GP pins = V_{SS}
Output Low-Voltage							
D080	GPIO	V_{OL}	—	—	0.6	V	$I_{OL} = 8.0\text{ mA}$ $V_{DD} = 4.5V$
	INT	V_{OL}	—	—	0.6	V	$I_{OL} = 1.6\text{ mA}$ $V_{DD} = 4.5V$
	SO, SDA	V_{OL}	—	—	0.6	V	$I_{OL} = 3.0\text{ mA}$ $V_{DD} = 1.8V$
	SDA	V_{OL}	—	—	0.8	V	$I_{OL} = 3.0\text{ mA}$ $V_{DD} = 4.5V$
Output High-Voltage							
D090	GPIO, INT, SO	V_{OH}	$V_{DD} - 0.7$	—	—	V	$I_{OH} = -3.0\text{ mA}$ $V_{DD} = 4.5V$
			$V_{DD} - 0.7$	—	—	V	$I_{OH} = -400\text{ }\mu A$ $V_{DD} = 1.8V$
Capacitive Loading Specs on Output Pins							
D101	GPIO, SO, INT	C_{IO}	—	—	50	pF	
D102	SDA	C_B	—	—	400	pF	

Note 1: This parameter is characterized, not 100% tested.

MCP23017/MCP23S17

1.2 AC Characteristics

FIGURE 1-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

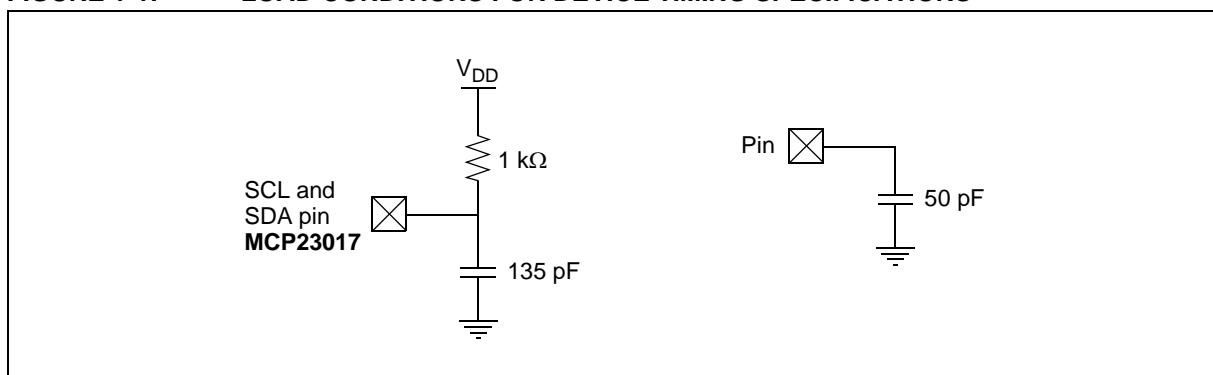


FIGURE 1-2: RESET AND DEVICE RESET TIMER TIMING

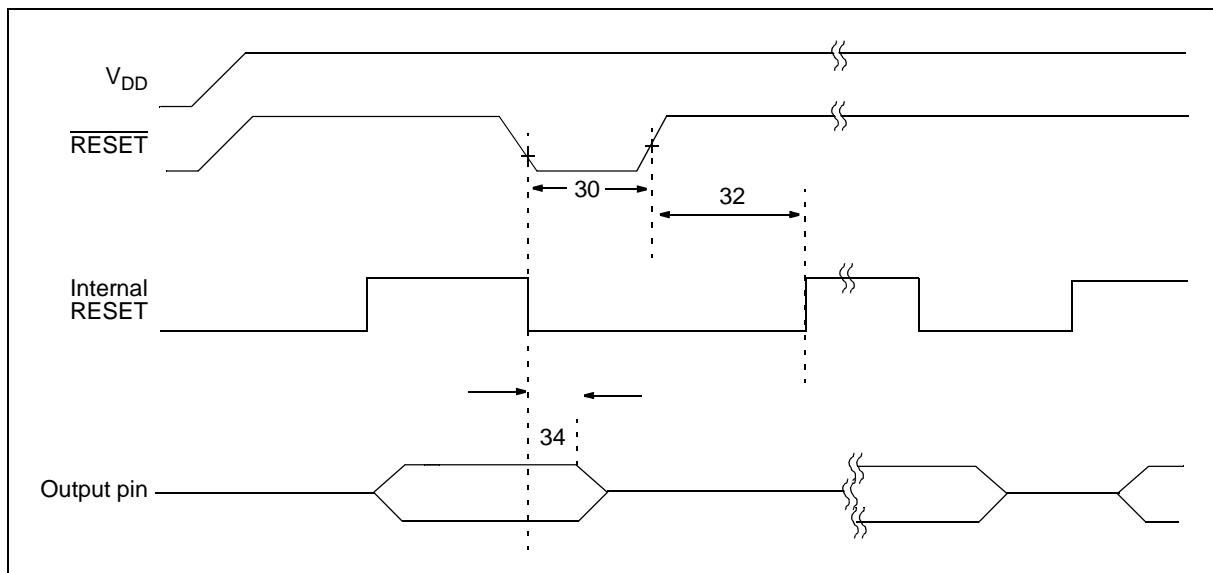


TABLE 1-2: DEVICE RESET SPECIFICATIONS

AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$							
Param. No.	Characteristic	Sym.	Min.	Typ. (1)	Max.	Units	Conditions
30	RESET Pulse Width (Low)	T_{RSTL}	1	—	—	μs	
32	Device Active After Reset high	T_{HLD}	—	0	—	ns	$V_{DD} = 5.0V$
34	Output High-Impedance From RESET Low	T_{IOZ}	—	—	1	μs	

Note 1: This parameter is characterized, not 100% tested.

MCP23017/MCP23S17

FIGURE 1-3: I²C BUS START/STOP BITS TIMING

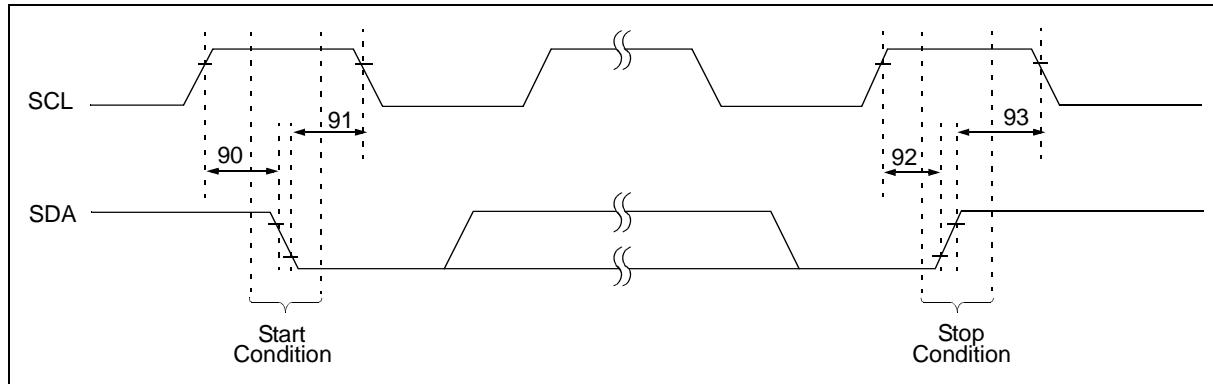


FIGURE 1-4: I²C BUS DATA TIMING

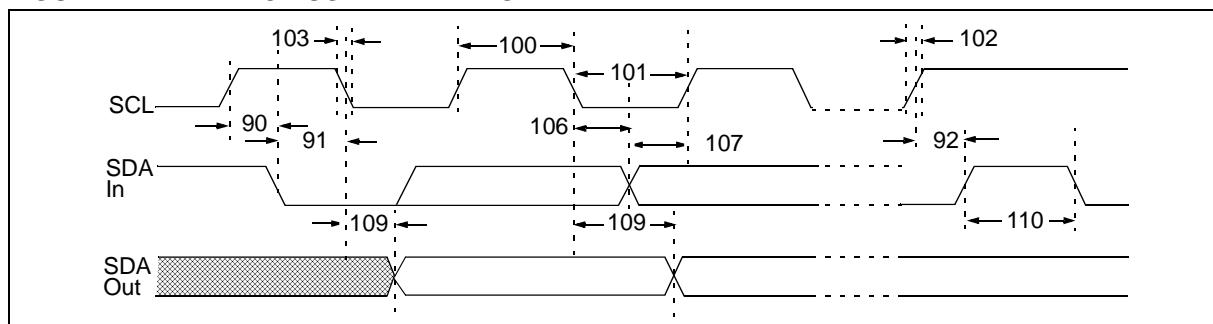


TABLE 1-3: I²C BUS DATA REQUIREMENTS

I²C Interface AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$, R_{PU} (SCL, SDA) = $1\text{ k}\Omega$, C_L (SCL, SDA) = 135 pF

Param. No.	Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions
100	Clock High Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T_{HIGH}					
			4.0	—	—	μs	$1.8V - 5.5V$
			0.6	—	—	μs	$2.7V - 5.5V$
			0.12	—	—	μs	$4.5V - 5.5V$
101	Clock Low Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T_{LOW}					
			4.7	—	—	μs	$1.8V - 5.5V$
			1.3	—	—	μs	$2.7V - 5.5V$
			0.32	—	—	μs	$4.5V - 5.5V$
102	SDA and SCL Rise Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T_R ⁽¹⁾					
			—	—	1000	ns	$1.8V - 5.5V$
			$20 + 0.1 C_B$ ⁽²⁾	—	300	ns	$2.7V - 5.5V$
			20	—	160	ns	$4.5V - 5.5V$
103	SDA and SCL Fall Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T_F ⁽¹⁾					
			—	—	300	ns	$1.8V - 5.5V$
			$20 + 0.1 C_B$ ⁽²⁾	—	300	ns	$2.7V - 5.5V$
			20	—	80	ns	$4.5V - 5.5V$

Note 1: This parameter is characterized, not 100% tested.

2: C_B is specified to be from 10 to 400 pF.

MCP23017/MCP23S17

TABLE 1-3: I²C BUS DATA REQUIREMENTS (CONTINUED)

I²C Interface AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$, R_{PU} (SCL, SDA) = 1 kΩ, C_L (SCL, SDA) = 135 pF

Param. No.	Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions
90	START Condition Setup Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{SU:STA}	4.7	—	—	μs	1.8V – 5.5V
			0.6	—	—	μs	2.7V – 5.5V
			0.16	—	—	μs	4.5V – 5.5V
91	START Condition Hold Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{HD:STA}	4.0	—	—	μs	1.8V – 5.5V
			0.6	—	—	μs	2.7V – 5.5V
			0.16	—	—	μs	4.5V – 5.5V
106	Data Input Hold Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{HD:DAT}	0	—	3.45	μs	1.8V – 5.5V
			0	—	0.9	μs	2.7V – 5.5V
			0	—	0.15	μs	4.5V – 5.5V
107	Data Input Setup Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{SU:DAT}	250	—	—	ns	1.8V – 5.5V
			100	—	—	ns	2.7V – 5.5V
			0.01	—	—	μs	4.5V – 5.5V
92	Stop Condition Setup Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{SU:STO}	4.0	—	—	μs	1.8V – 5.5V
			0.6	—	—	μs	2.7V – 5.5V
			0.16	—	—	μs	4.5V – 5.5V
109	Output Valid From Clock: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{AA}	—	—	3.45	μs	1.8V – 5.5V
			—	—	0.9	μs	2.7V – 5.5V
			—	—	0.18	μs	4.5V – 5.5V
110	Bus Free Time: 100 kHz mode 400 kHz mode 1.7 MHz mode	T _{BUF}	4.7	—	—	μs	1.8V – 5.5V
			1.3	—	—	μs	2.7V – 5.5V
			N/A	—	N/A	μs	4.5V – 5.5V
111	Bus Capacitive Loading: 100 kHz and 400 kHz 1.7 MHz	C _B	—	—	400	pF	Note 1
			—	—	100	pF	Note 1
112	Input Filter Spike Suppression (SDA and SCL): 100 kHz and 400 kHz 1.7 MHz	T _{SP}	—	—	50	ns	
			—	—	10	ns	Spike suppression off

Note 1: This parameter is characterized, not 100% tested.

2: C_B is specified to be from 10 to 400 pF.

MCP23017/MCP23S17

FIGURE 1-5: SPI INPUT TIMING

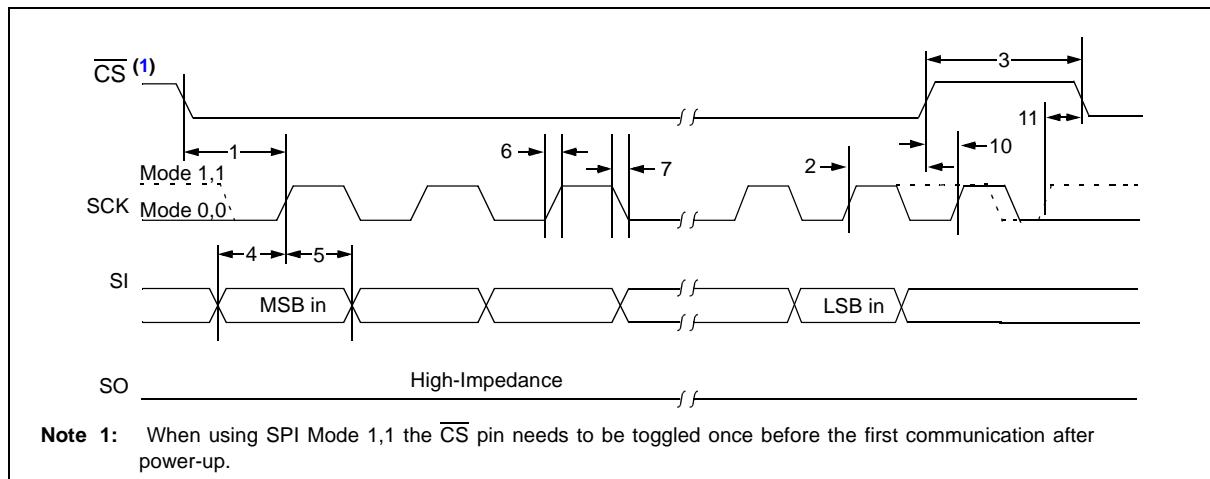


FIGURE 1-6: SPI OUTPUT TIMING

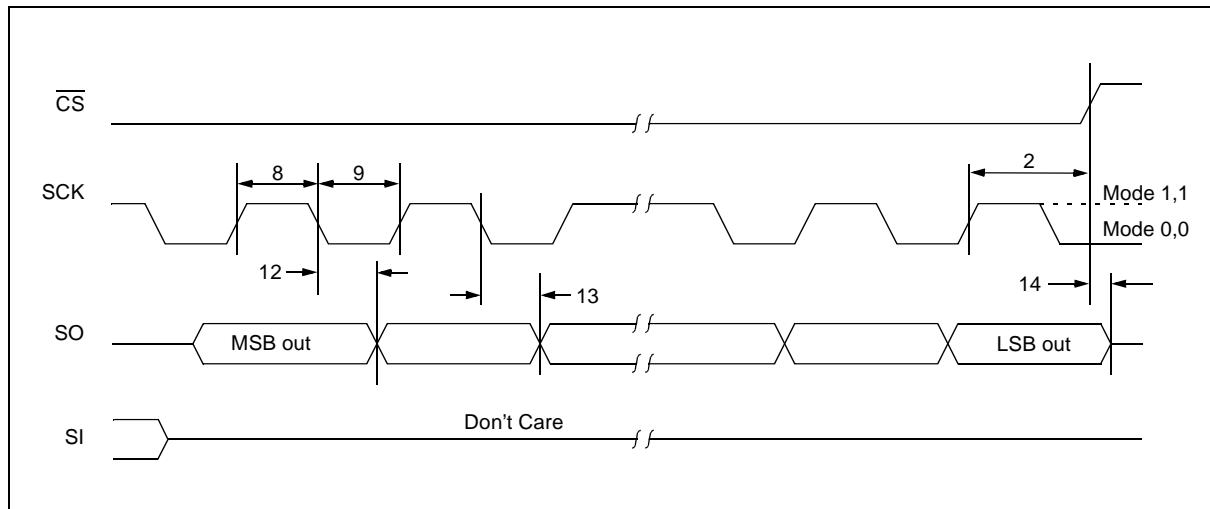


TABLE 1-4: SPI INTERFACE REQUIREMENTS

SPI Interface AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$							
Param. No.	Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions
—	Clock Frequency	F_{CLK}	—	—	5	MHz	$1.8V - 5.5V$
			—	—	10	MHz	$2.7V - 5.5V$
			—	—	10	MHz	$4.5V - 5.5V$
1	CS Setup Time	T_{CSS}	50	—	—	ns	
2	CS Hold Time	T_{CSH}	100	—	—	ns	$1.8V - 5.5V$
			50	—	—	ns	$2.7V - 5.5V$
3	CS Disable Time	T_{CSD}	100	—	—	ns	$1.8V - 5.5V$
			50	—	—	ns	$2.7V - 5.5V$
4	Data Setup Time	T_{SU}	20	—	—	ns	$1.8V - 5.5V$
			10	—	—	ns	$2.7V - 5.5V$

Note 1: This parameter is characterized, not 100% tested.

MCP23017/MCP23S17

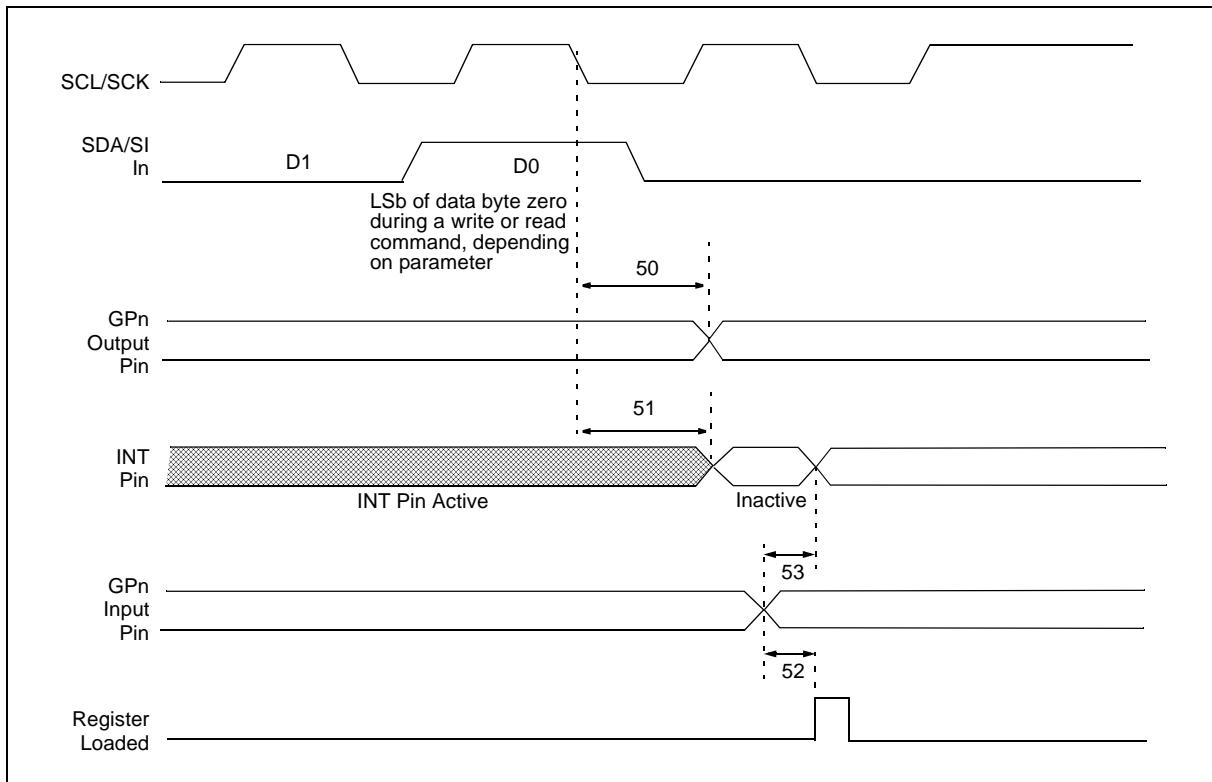
TABLE 1-4: SPI INTERFACE REQUIREMENTS (CONTINUED)

SPI Interface AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$

Param. No.	Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions
5	Data Hold Time	T_{HD}	20	—	—	ns	$1.8V - 5.5V$
			10	—	—	ns	$2.7V - 5.5V$
6	CLK Rise Time	T_R	—	—	2	μs	Note 1
7	CLK Fall Time	T_F	—	—	2	μs	Note 1
8	Clock High Time	T_{HI}	90	—	—	ns	$1.8V - 5.5V$
			45	—	—	ns	$2.7V - 5.5V$
9	Clock Low Time	T_{LO}	90	—	—	ns	$1.8V - 5.5V$
			45	—	—	ns	$2.7V - 5.5V$
10	Clock Delay Time	T_{CLD}	50	—	—	ns	
11	Clock Enable Time	T_{CLE}	50	—	—	ns	
12	Output Valid from Clock Low	T_V	—	—	90	ns	$1.8V - 5.5V$
			—	—	45	ns	$2.7V - 5.5V$
13	Output Hold Time	T_{HO}	0	—	—	ns	
14	Output Disable Time	T_{DIS}	—	—	100	ns	

Note 1: This parameter is characterized, not 100% tested.

FIGURE 1-7: GPIO AND INT TIMING



MCP23017/MCP23S17

TABLE 1-5: GP AND INT PINS REQUIREMENTS

GP and INT Pins AC Characteristics: Unless otherwise noted, $1.8V \leq V_{DD} \leq 5.5V$ at $-40^{\circ}C \leq T_A \leq +125^{\circ}C$							
Param. No.	Characteristic	Sym.	Min.	Typ.	Max.	Units	Conditions
50	Serial Data to Output Valid	T_{GPOV}	—	—	500	ns	
51	Interrupt Pin Disable Time	T_{INTD}	—	—	600	ns	
52	GP Input Change to Register Valid	T_{GPIV}	—	—	450	ns	
53	IOC Event to INT Active	T_{GPINT}	—	—	600	ns	
	Glitch Filter on GP Pins	T_{GLITCH}	—	—	150	ns	Note 1

Note 1: This parameter is characterized, not 100% tested.

MCP23017/MCP23S17

2.0 PIN DESCRIPTIONS

The descriptions of the pins are listed in [Table 2-1](#).

TABLE 2-1: PINOUT DESCRIPTION

Pin Name	QFN	SOIC SPDIP SSOP	Pin Type	Function
GPB0	25	1	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB1	26	2	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB2	27	3	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB3	28	4	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB4	1	5	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB5	2	6	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB6	3	7	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPB7	4	8	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
V _{DD}	5	9	P	Power
V _{SS}	6	10	P	Ground
NC/CS	7	11	I	NC (MCP23017)/Chip Select (MCP23S17)
SCK	8	12	I	Serial clock input
SDA/SI	9	13	I/O	Serial data I/O (MCP23017)/Serial data input (MCP23S17)
NC/SO	10	14	O	NC (MCP23017)/Serial data out (MCP23S17)
A0	11	15	I	Hardware address pin. Must be externally biased.
A1	12	16	I	Hardware address pin. Must be externally biased.
A2	13	17	I	Hardware address pin. Must be externally biased.
RESET	14	18	I	Hardware reset. Must be externally biased.
INTB	15	19	O	Interrupt output for PORTB. Can be configured as active-high, active-low or open-drain.
INTA	16	20	O	Interrupt output for PORTA. Can be configured as active-high, active-low or open-drain.
GPA0	17	21	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA1	18	22	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA2	19	23	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA3	20	24	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA4	21	25	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA5	22	26	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA6	23	27	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
GPA7	24	28	I/O	Bidirectional I/O pin. Can be enabled for interrupt-on-change and/or internal weak pull-up resistor.
EP	29	—	—	Exposed Thermal Pad. Either connect to V _{SS} , or leave unconnected.

MCP23017/MCP23S17

3.0 DEVICE OVERVIEW

The MCP23017/MCP23S17 (MCP23X17) device family provides 16-bit, general purpose parallel I/O expansion for I²C bus or SPI applications. The two devices differ only in the serial interface:

- MCP23017 – I²C interface
- MCP23S17 – SPI interface

The MCP23X17 consists of multiple 8-bit configuration registers for input, output and polarity selection. The system master can enable the I/Os as either inputs or outputs by writing the I/O configuration bits (IODIRA/B). The data for each input or output is kept in the corresponding input or output register. The polarity of the Input Port register can be inverted with the Polarity Inversion register. All registers can be read by the system master.

The 16-bit I/O port functionally consists of two 8-bit ports (PORTA and PORTB). The MCP23X17 can be configured to operate in the 8-bit or 16-bit modes via IOCON.BANK.

There are two interrupt pins, INTA and INTB, that can be associated with their respective ports, or can be logically OR'ed together so that both pins will activate if either port causes an interrupt.

The interrupt output can be configured to activate under two conditions (mutually exclusive):

1. When any input state differs from its corresponding Input Port register state. This is used to indicate to the system master that an input state has changed.
2. When an input state differs from a preconfigured register value (DEFVAL register).

The Interrupt Capture register captures port values at the time of the interrupt, thereby saving the condition that caused the interrupt.

The Power-on Reset (POR) sets the registers to their default values and initializes the device state machine.

The hardware address pins are used to determine the device address.

3.1 Power-on Reset (POR)

The on-chip POR circuit holds the device in reset until V_{DD} has reached a high enough voltage to deactivate the POR circuit (i.e., release the device from reset). The maximum V_{DD} rise time is specified in [Section 1.0 “Electrical Characteristics”](#).

When the device exits the POR condition (releases reset), device operating parameters (i.e., voltage, temperature, serial bus frequency, etc.) must be met to ensure proper operation.

3.2 Serial Interface

This block handles the functionality of the I²C ([MCP23017](#)) or SPI ([MCP23S17](#)) interface protocol. The MCP23X17 contains 22 individual registers (11 register pairs) that can be addressed through the Serial Interface block, as shown in [Table 3-1](#).

TABLE 3-1: REGISTER ADDRESSES

Address IOCON.BANK = 1	Address IOCON.BANK = 0	Access to:
00h	00h	IODIRA
10h	01h	IODIRB
01h	02h	IPOLA
11h	03h	IPOLB
02h	04h	GPINTENA
12h	05h	GPINTENB
03h	06h	DEFVALA
13h	07h	DEFVALB
04h	08h	INTCONA
14h	09h	INTCONB
05h	0Ah	IOCON
15h	0Bh	IOCON
06h	0Ch	GPPUA
16h	0Dh	GPPUB
07h	0Eh	INTFA
17h	0Fh	INTFB
08h	10h	INTCAPA
18h	11h	INTCAPB
09h	12h	GPIOA
19h	13h	GPIOB
0Ah	14h	OLATA
1Ah	15h	OLATB

3.2.1 BYTE MODE AND SEQUENTIAL MODE

The MCP23X17 family has the ability to operate in Byte mode or Sequential mode (IOCON.SEQOP).

Byte mode disables automatic Address Pointer incrementing. When operating in Byte mode, the MCP23X17 family does not increment its internal address counter after each byte during the data transfer. This gives the ability to continually access the same address by providing extra clocks (without additional control bytes). This is useful for polling the GPIO register for data changes or for continually writing to the output latches.

A special mode (**Byte mode with IOCON.BANK = 0**) causes the address pointer to toggle between associated A/B register pairs. For example, if the BANK bit is cleared and the Address Pointer is initially set to address 12h (GPIOA) or 13h (GPIOB), the pointer will toggle between GPIOA and GPIOB. Note that the Address Pointer can initially point to either address in the register pair.

Sequential mode enables automatic address pointer incrementing. When operating in Sequential mode, the MCP23X17 family increments its address counter after each byte during the data transfer. The Address Pointer automatically rolls over to address 00h after accessing the last register.

These two modes are not to be confused with single writes/reads and continuous writes/reads that are serial protocol sequences. For example, the device may be configured for Byte mode and the master may perform a continuous read. In this case, the MCP23X17 would not increment the Address Pointer and would repeatedly drive data from the same location.

3.2.2 I²C INTERFACE

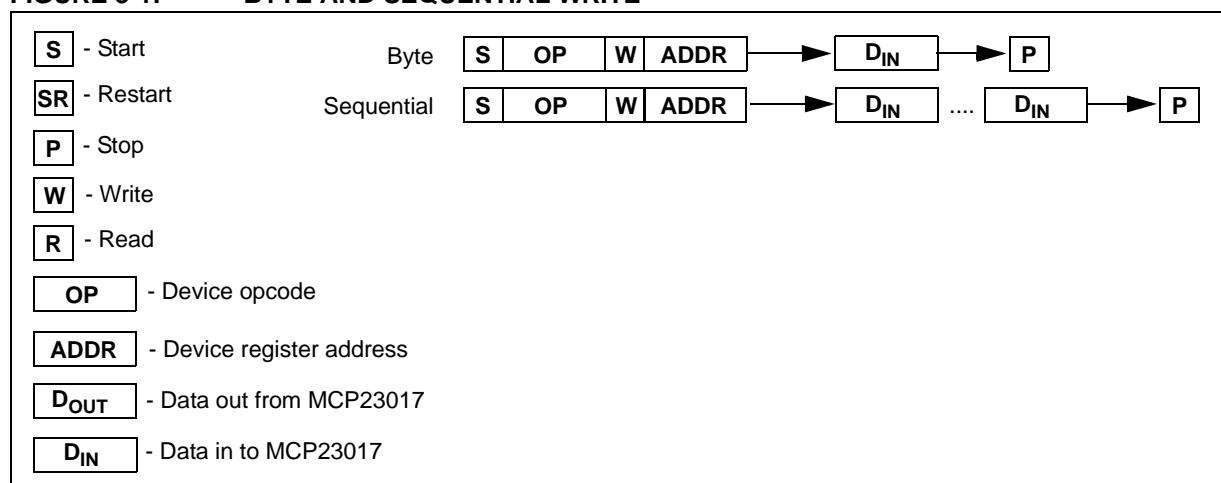
3.2.2.1 I²C Write Operation

The I²C write operation includes the control byte and register address sequence, as shown in [Figure 3-1](#). This sequence is followed by eight bits of data from the master and an Acknowledge (ACK) from the MCP23017. The operation is ended with a Stop (P) or Restart (SR) condition being generated by the master.

Data is written to the MCP23017 after every byte transfer. If a Stop or Restart condition is generated during a data transfer, the data will not be written to the MCP23017.

Both “byte writes” and “sequential writes” are supported by the MCP23017. If Sequential mode is enabled (IOCON, SEQOP = 0) (default), the MCP23017 increments its address counter after each ACK during the data transfer.

FIGURE 3-1: BYTE AND SEQUENTIAL WRITE

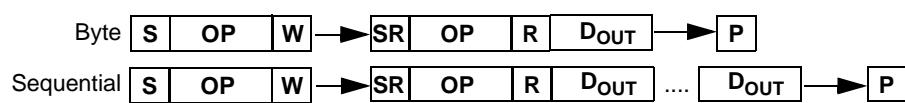


MCP23017/MCP23S17

3.2.2.2 I²C Read Operation

I²C Read operations include the control byte sequence, as shown in [Figure 3-2](#). This sequence is followed by another control byte (including the Start condition and ACK) with the R/W bit set (R/W = 1). The MCP23017 then transmits the data contained in the addressed register. The sequence is ended with the master generating a Stop or Restart condition.

FIGURE 3-2: BYTE AND SEQUENTIAL READ



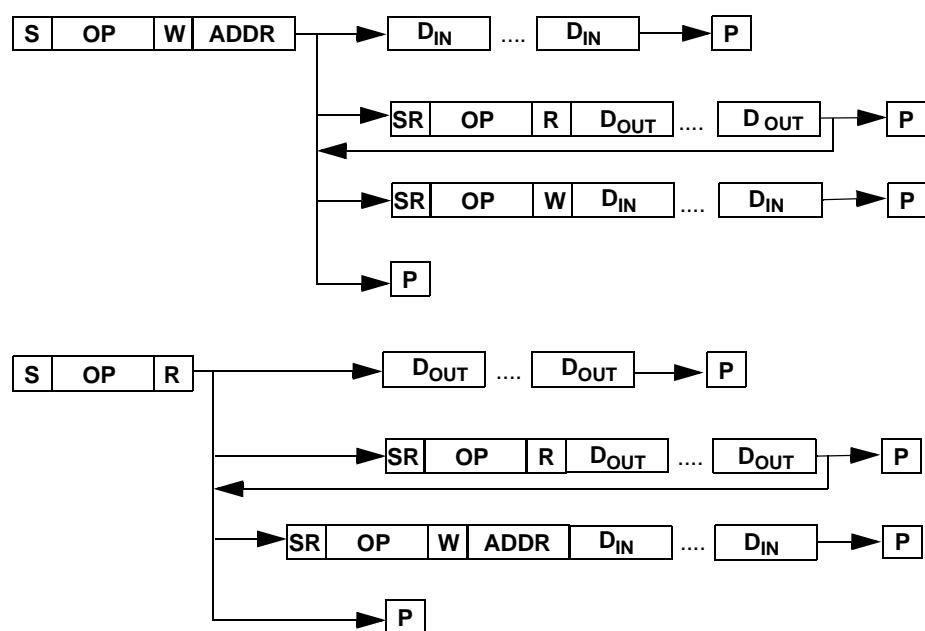
3.2.2.3 I²C Sequential Write/Read

For sequential operations (Write or Read), instead of transmitting a Stop or Restart condition after the data transfer, the master clocks the next byte pointed to by the address pointer (see [Section 3.2.1 "Byte Mode and Sequential Mode"](#) for details regarding sequential operation control).

The sequence ends with the master sending a Stop or Restart condition.

The MCP23017 Address Pointer will roll over to address zero after reaching the last register address. Refer to [Figure 3-3](#).

FIGURE 3-3: MCP23017 I²C DEVICE PROTOCOL



MCP23017/MCP23S17

3.2.3 SPI INTERFACE

3.2.3.1 SPI Write Operation

The SPI write operation is started by lowering \overline{CS} . The Write command (slave address with R/W bit cleared) is then clocked into the device. The opcode is followed by an address and at least one data byte.

3.2.3.2 SPI Read Operation

The SPI read operation is started by lowering \overline{CS} . The SPI read command (slave address with R/W bit set) is then clocked into the device. The opcode is followed by an address, with at least one data byte being clocked out of the device.

3.2.3.3 SPI Sequential Write/Read

For sequential operations, instead of deselecting the device by raising CS , the master clocks the next byte pointed to by the Address Pointer. (see [Section 3.2.1 "Byte Mode and Sequential Mode"](#) for details regarding sequential operation control).

The sequence ends by the raising of CS .

The MCP23S17 Address Pointer will roll over to address zero after reaching the last register address.

3.3 Hardware Address Decoder

The hardware address pins are used to determine the device address. To address a device, the corresponding address bits in the control byte must match the pin state. The pins must be biased externally.

3.3.1 ADDRESSING I²C DEVICES (MCP23017)

The MCP23017 is a slave I²C interface device that supports 7-bit slave addressing, with the read/write bit filling out the control byte. The slave address contains

four fixed bits and three user-defined hardware address bits (pins A2, A1 and A0). [Figure 3-4](#) shows the control byte format.

3.3.2 ADDRESSING SPI DEVICES (MCP23S17)

The MCP23S17 is a slave SPI device. The slave address contains four fixed bits and three user-defined hardware address bits (if enabled via IOCON.HAEN) (pins A2, A1 and A0) with the read/write bit filling out the control byte. [Figure 3-5](#) shows the control byte format. The address pins should be externally biased even if disabled (IOCON.HAEN = 0).

FIGURE 3-4: I²C CONTROL BYTE FORMAT

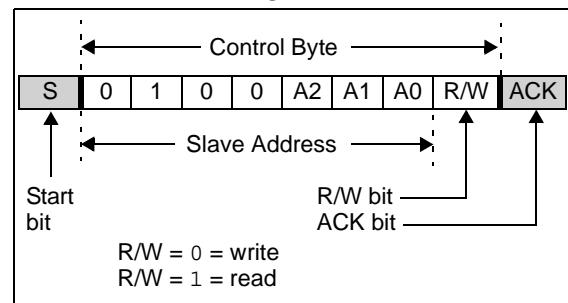


FIGURE 3-5: SPI CONTROL BYTE FORMAT

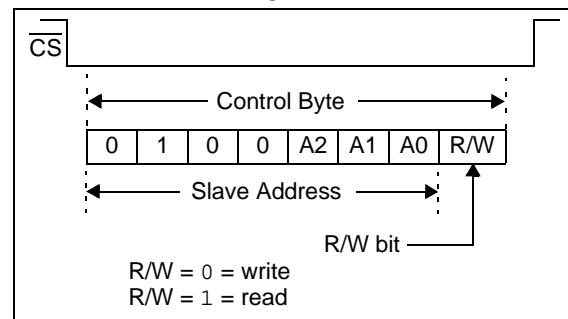
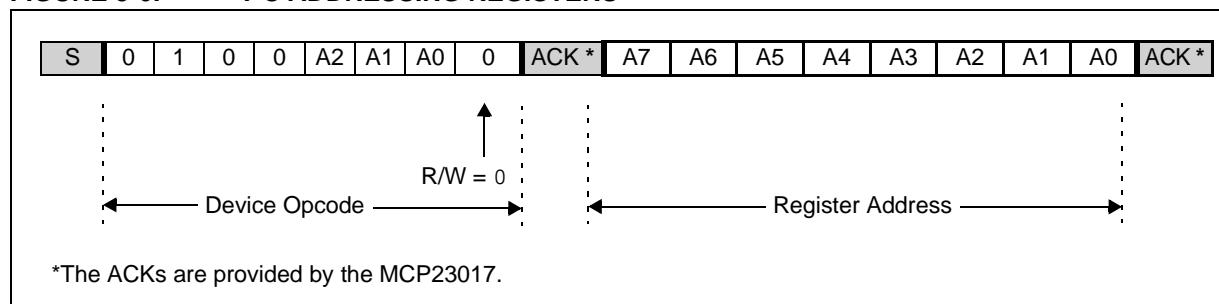


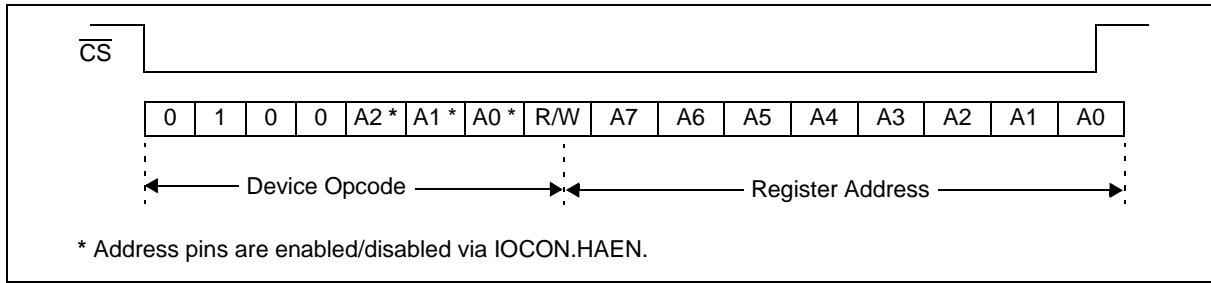
FIGURE 3-6: I²C ADDRESSING REGISTERS



*The ACKs are provided by the MCP23017.

MCP23017/MCP23S17

FIGURE 3-7: SPI ADDRESSING REGISTERS



3.4 GPIO Port

The GPIO module is a general purpose, 16-bit wide, bidirectional port that is functionally split into two 8-bit wide ports.

The GPIO module contains the data ports (GPIO_n), internal pull-up resistors and the output latches (OLAT_n).

Reading the GPIO_n register reads the value on the port. Reading the OLAT_n register only reads the latches, not the actual value on the port.

Writing to the GPIO_n register actually causes a write to the latches (OLAT_n). Writing to the OLAT_n register forces the associated output drivers to drive to the level in OLAT_n. Pins configured as inputs turn off the associated output driver and put it in high-impedance.

TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE GPIO PORTS (BANK = 1)

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
IODIRA	00	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLA	01	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENA	02	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPPUA	06	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPIOA	09	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATA	0A	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000
IODIRB	10	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLB	11	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENB	12	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPPUB	16	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPIOB	19	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATB	1A	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000

TABLE 3-3: SUMMARY OF REGISTERS ASSOCIATED WITH THE GPIO PORTS (BANK = 0)

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
IODIRA	00	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IODIRB	01	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLA	02	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
IPOLB	03	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENA	04	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPINTENB	05	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPPUA	0C	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPPUB	0D	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPIOA	12	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
GPIOB	13	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATA	14	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000
OLATB	15	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000

MCP23017/MCP23S17

3.5 Configuration and Control Registers

There are 21 registers associated with the MCP23X17, as shown in [Tables 3-4](#) and [3-5](#). The two tables show the register mapping with the two BANK bit values. Ten registers are associated with PORTA and ten are

associated with PORTB. One register (IOCON) is shared between the two ports. The PORTA registers are identical to the PORTB registers, therefore, they will be referred to without differentiating between the port designation (i.e., they will not have the "A" or "B" designator assigned) in the register tables.

TABLE 3-4: CONTROL REGISTER SUMMARY (IOCON.BANK = 1)

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
IODIRA	00	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLA	01	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENA	02	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
DEFVALA	03	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
INTCONA	04	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
IOCON	05	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
GPPUA	06	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
INTFA	07	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	0000 0000
INTCAPA	08	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
GPIOA	09	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATA	0A	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000
IODIRB	10	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLB	11	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENB	12	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
DEFVALB	13	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
INTCONB	14	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
IOCON	15	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
GPPUB	16	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
INTFB	17	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	0000 0000
INTCAPB	18	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
GPIOB	19	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATB	1A	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000

TABLE 3-5: CONTROL REGISTER SUMMARY (IOCON.BANK = 0)

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
IODIRA	00	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IODIRB	01	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0	1111 1111
IPOLA	02	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
IPOLB	03	IP7	IP6	IP5	IP4	IP3	IP2	IP1	IP0	0000 0000
GPINTENA	04	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
GPINTENB	05	GPINT7	GPINT6	GPINT5	GPINT4	GPINT3	GPINT2	GPINT1	GPINT0	0000 0000
DEFVALA	06	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
DEFVALB	07	DEF7	DEF6	DEF5	DEF4	DEF3	DEF2	DEF1	DEF0	0000 0000
INTCONA	08	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
INTCONB	09	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	0000 0000
IOCON	0A	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
IOCON	0B	BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—	0000 0000
GPPUA	0C	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000
GPPUB	0D	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	0000 0000

MCP23017/MCP23S17

TABLE 3-5: CONTROL REGISTER SUMMARY (IOCON.BANK = 0) (CONTINUED)

Register Name	Address (hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	POR/RST value
INTFA	0E	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	0000 0000
INTFB	0F	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	0000 0000
INTCAPA	10	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
INTCAPB	11	ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0	0000 0000
GPIOA	12	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
GPIOB	13	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0	0000 0000
OLATA	14	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000
OLATB	15	OL7	OL6	OL5	OL4	OL3	OL2	OL1	OL0	0000 0000

3.5.1 I/O DIRECTION REGISTER

Controls the direction of the data I/O.

When a bit is set, the corresponding pin becomes an input. When a bit is clear, the corresponding pin becomes an output.

REGISTER 3-1: IODIR: I/O DIRECTION REGISTER (ADDR 0x00)

| R/W-1 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| IO7 | IO6 | IO5 | IO4 | IO3 | IO2 | IO1 | IO0 |
| bit 7 | bit 0 | | | | | | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **IO<7:0>**: Controls the direction of data I/O <7:0>

1 = Pin is configured as an input.

0 = Pin is configured as an output.

3.5.2 INPUT POLARITY REGISTER

This register allows the user to configure the polarity on the corresponding GPIO port bits.

If a bit is set, the corresponding GPIO register bit will reflect the inverted value on the pin.

REGISTER 3-2: IPOL: INPUT POLARITY PORT REGISTER (ADDR 0x01)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| IP7 | IP6 | IP5 | IP4 | IP3 | IP2 | IP1 | IP0 |
| bit 7 | bit 0 | | | | | | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **IP<7:0>**: Controls the polarity inversion of the input pins <7:0>

1 = GPIO register bit reflects the opposite logic state of the input pin.

0 = GPIO register bit reflects the same logic state of the input pin.

MCP23017/MCP23S17

3.5.3 INTERRUPT-ON-CHANGE CONTROL REGISTER

The GPINTEN register controls the interrupt-on-change feature for each pin.

If a bit is set, the corresponding pin is enabled for interrupt-on-change. The DEFVAL and INTCON registers must also be configured if any pins are enabled for interrupt-on-change.

REGISTER 3-3: GPINTEN: INTERRUPT-ON-CHANGE PINS (ADDR 0x02) ([Note 1](#))

| R/W-0 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| GPINT7 | GPINT6 | GPINT5 | GPINT4 | GPINT3 | GPINT2 | GPINT1 | GPINT0 |
| bit 7 | bit 0 | | | | | | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **GPINT<7:0>**: General purpose I/O interrupt-on-change bits <7:0>

1 = Enables GPIO input pin for interrupt-on-change event.

0 = Disables GPIO input pin for interrupt-on-change event.

Note 1: Refer to INTCON.

3.5.4 DEFAULT COMPARE REGISTER FOR INTERRUPT-ON-CHANGE

The default comparison value is configured in the DEFVAL register. If enabled (via GPINTEN and INTCON) to compare against the DEFVAL register, an opposite value on the associated pin will cause an interrupt to occur.

REGISTER 3-4: DEFVAL: DEFAULT VALUE REGISTER (ADDR 0x03)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| DEF7 | DEF6 | DEF5 | DEF4 | DEF3 | DEF2 | DEF1 | DEF0 |
| bit 7 | bit 0 | | | | | | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **DEF<7:0>**: Sets the compare value for pins configured for interrupt-on-change from defaults <7:0> ([Note 1](#))

If the associated pin level is the opposite from the register bit, an interrupt occurs. ([Note 2](#))

Note 1: Refer to INTCON.

2: Refer to INTCON and GPINTEN.

MCP23017/MCP23S17

3.5.5 INTERRUPT CONTROL REGISTER

The INTCON register controls how the associated pin value is compared for the interrupt-on-change feature. If a bit is set, the corresponding I/O pin is compared against the associated bit in the DEFVAL register. If a bit value is clear, the corresponding I/O pin is compared against the previous value.

REGISTER 3-5: INTCON: INTERRUPT-ON-CHANGE CONTROL REGISTER (ADDR 0x04) (Note 1)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| IOC7 | IOC6 | IOC5 | IOC4 | IOC3 | IOC2 | IOC1 | IOC0 |
| bit 7 | bit 0 | | | | | | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **IOC<7:0>**: Controls how the associated pin value is compared for interrupt-on-change <7:0>
1 = Pin value is compared against the associated bit in the DEFVAL register.
0 = Pin value is compared against the previous pin value.

Note 1: Refer to INTCON and GPINTEN.

3.5.6 CONFIGURATION REGISTER

The IOCON register contains several bits for configuring the device:

The BANK bit changes how the registers are mapped (see [Tables 3-4](#) and [3-5](#) for more details).

- If BANK = 1, the registers associated with each port are segregated. Registers associated with PORTA are mapped from address 00h - 0Ah and registers associated with PORTB are mapped from 10h - 1Ah.
- If BANK = 0, the A/B registers are paired. For example, IODIRA is mapped to address 00h and IODIRB is mapped to the next address (address 01h). The mapping for all registers is from 00h - 15h.

It is important to take care when changing the BANK bit as the address mapping changes after the byte is clocked into the device. The address pointer may point to an invalid location after the bit is modified.

For example, if the device is configured to automatically increment its internal Address Pointer, the following scenario would occur:

- BANK = 0
- Write 80h to address 0Ah (IOCON) to set the BANK bit
- Once the write completes, the internal address now points to 0Bh which is an invalid address when the BANK bit is set.

For this reason, when changing the BANK bit, it is advised to only perform byte writes to this register.

The MIRROR bit controls how the INTA and INTB pins function with respect to each other.

- When MIRROR = 1, the INTn pins are functionally OR'ed so that an interrupt on either port will cause both pins to activate.
- When MIRROR = 0, the INT pins are separated. Interrupt conditions on a port will cause its respective INT pin to activate.

The Sequential Operation (**SEQOP**) controls the incrementing function of the Address Pointer. If the address pointer is disabled, the Address Pointer does not automatically increment after each byte is clocked during a serial transfer. This feature is useful when it is desired to continuously poll (read) or modify (write) a register.

The Slew Rate (**DISSLW**) bit controls the slew rate function on the SDA pin. If enabled, the SDA slew rate will be controlled when driving from a high to low.

The Hardware Address Enable (**HAEN**) bit enables/disables hardware addressing on the MCP23S17 only. The address pins (A2, A1 and A0) must be externally biased, regardless of the HAEN bit value.

If enabled (HAEN = 1), the device's hardware address matches the address pins.

If disabled (HAEN = 0), the device's hardware address is A2 = A1 = A0 = 0.

MCP23017/MCP23S17

The Open-Drain (**ODR**) control bit enables/disables the INT pin for open-drain configuration. Setting this bit overrides the INTPOL bit.

The Interrupt Polarity (**INTPOL**) sets the polarity of the INT pin. This bit is functional only when the ODR bit is cleared, configuring the INT pin as active push-pull.

REGISTER 3-6: IOCON: I/O EXPANDER CONFIGURATION REGISTER (ADDR 0x05)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
BANK	MIRROR	SEQOP	DISSLW	HAEN	ODR	INTPOL	—
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 7 **BANK:** Controls how the registers are addressed
1 = The registers associated with each port are separated into different banks.
0 = The registers are in the same bank (addresses are sequential).
- bit 6 **MIRROR:** INT Pins Mirror bit
1 = The INT pins are internally connected
0 = The INT pins are not connected. INTA is associated with PORTA and INTB is associated with PORTB
- bit 5 **SEQOP:** Sequential Operation mode bit
1 = Sequential operation disabled, address pointer does not increment.
0 = Sequential operation enabled, address pointer increments.
- bit 4 **DISSLW:** Slew Rate control bit for SDA output
1 = Slew rate disabled
0 = Slew rate enabled
- bit 3 **HAEN:** Hardware Address Enable bit (MCP23S17 only) ([Note 1](#))
1 = Enables the MCP23S17 address pins.
0 = Disables the MCP23S17 address pins.
- bit 2 **ODR:** Configures the INT pin as an open-drain output
1 = Open-drain output (overrides the INTPOL bit.)
0 = Active driver output (INTPOL bit sets the polarity.)
- bit 1 **INTPOL:** This bit sets the polarity of the INT output pin
1 = Active-high
0 = Active-low
- bit 0 **Unimplemented:** Read as '0'

Note 1: Address pins are always enabled on the MCP23017.

MCP23017/MCP23S17

3.5.7 PULL-UP RESISTOR CONFIGURATION REGISTER

The GPPU register controls the pull-up resistors for the port pins. If a bit is set and the corresponding pin is configured as an input, the corresponding port pin is internally pulled up with a 100 kΩ resistor.

REGISTER 3-7: GPPU: GPIO PULL-UP RESISTOR REGISTER (ADDR 0x06)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PU7 | PU6 | PU5 | PU4 | PU3 | PU2 | PU1 | PU0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

bit 7-0 **PU<7:0>** Controls the weak pull-up resistors on each pin (when configured as an input)

1 = Pull-up enabled
0 = Pull-up disabled

3.5.8 INTERRUPT FLAG REGISTER

The INTF register reflects the interrupt condition on the port pins of any pin that is enabled for interrupts via the GPINTEN register. A set bit indicates that the associated pin caused the interrupt.

This register is read-only. Writes to this register will be ignored.

REGISTER 3-8: INTF: INTERRUPT FLAG REGISTER (ADDR 0x07)

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

bit 7-0 **INT<7:0>**: Reflects the interrupt condition on the port. It reflects the change only if interrupts are enabled per GPINTEN<7:0>.

1 = Pin caused interrupt.
0 = Interrupt not pending

MCP23017/MCP23S17

3.5.9 INTERRUPT CAPTURED REGISTER

The INTCAP register captures the GPIO port value at the time the interrupt occurred. The register is read-only and is updated only when an interrupt occurs. The register remains unchanged until the interrupt is cleared via a read of INTCAP or GPIO.

REGISTER 3-9: INTCAP: INTERRUPT CAPTURED VALUE FOR PORT REGISTER (ADDR 0x08)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
ICP7	ICP6	ICP5	ICP4	ICP3	ICP2	ICP1	ICP0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ICP<7:0>**: Reflects the logic level on the port pins at the time of interrupt due to pin change <7:0>

1 = Logic-high

0 = Logic-low

3.5.10 PORT REGISTER

The GPIO register reflects the value on the port. Reading from this register reads the port. Writing to this register modifies the Output Latch (OLAT) register.

REGISTER 3-10: GPIO: GENERAL PURPOSE I/O PORT REGISTER (ADDR 0x09)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| GP7 | GP6 | GP5 | GP4 | GP3 | GP2 | GP1 | GP0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **GP<7:0>**: Reflects the logic level on the pins <7:0>

1 = Logic-high

0 = Logic-low

MCP23017/MCP23S17

3.5.11 OUTPUT LATCH REGISTER (OLAT)

The OLAT register provides access to the output latches. A read from this register results in a read of the OLAT and not the port itself. A write to this register modifies the output latches that modifies the pins configured as outputs.

REGISTER 3-11: OLAT: OUTPUT LATCH REGISTER 0 (ADDR 0x0A)

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| OL7 | OL6 | OL5 | OL4 | OL3 | OL2 | OL1 | OL0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

bit 7-0 **OL<7:0>**: Reflects the logic level on the output latch <7:0>

1 = Logic-high
0 = Logic-low

3.6 Interrupt Logic

If enabled, the MCP23X17 activates the INTn interrupt output when one of the port pins changes state or when a pin does not match the preconfigured default. Each pin is individually configurable as follows:

- Enable/disable interrupt via GPINTEN
- Can interrupt on either pin change or change from default as configured in DEFVAL

Both conditions are referred to as Interrupt-on-Change (IOC).

The interrupt control module uses the following registers/bits:

- IOCON.MIRROR – controls if the two interrupt pins mirror each other
- GPINTEN – Interrupt enable register
- INTCON – controls the source for the IOC
- DEFVAL – contains the register default for IOC operation

3.6.1 INTA AND INTB

There are two interrupt pins: INTA and INTB. By default, INTA is associated with GPAn pins (PORTA) and INTB is associated with GPBn pins (PORTB). Each port has an independent signal which is cleared if its associated GPIO or INTCAP register is read.

3.6.1.1 Mirroring the INT pins

Additionally, the INTn pins can be configured to mirror each other so that any interrupt will cause both pins to go active. This is controlled via IOCON.MIRROR.

If IOCON.MIRROR = 0, the internal signals are routed independently to the INTA and INTB pads.

If IOCON.MIRROR = 1, the internal signals are OR'ed together and routed to the INTn pads. In this case, the interrupt will only be cleared if the associated GPIO or INTCAP is read (see [Table 3-6](#)).

TABLE 3-6: INTERRUPT OPERATION (IOCON.MIRROR = 1)

Interrupt Condition	Read PORTn ⁽¹⁾	Interrupt Result
GPIOA	PORTA	Clear
	PORTB	Unchanged
GPIOB	PORTA	Unchanged
	PORTB	Clear
GPIOA and GPIOB	PORTA	Unchanged
	PORTB	Unchanged
	Both PORTA and PORTB	Clear

Note 1: PORTn = GPIOn or INTCAPn

3.6.2 IOC FROM PIN CHANGE

If enabled, the MCP23X17 generates an interrupt if a mismatch condition exists between the current port value and the previous port value. Only IOC-enabled pins will be compared. Refer to [Registers 3-3](#) and [3-5](#).

3.6.3 IOC FROM REGISTER DEFAULT

If enabled, the MCP23X17 generates an interrupt if a mismatch occurs between the DEFVAL register and the port. Only IOC enabled pins are compared. Refer to [Registers 3-3](#), [3-4](#) and [3-5](#).

3.6.4 INTERRUPT OPERATION

The INTn interrupt output can be configured as active-low, active-high or open-drain via the IOCON register.

Only those pins that are configured as an input (IODIR register) with Interrupt-On-Change (IOC) enabled (IOINTEN register) can cause an interrupt. Pins defined as an output have no effect on the interrupt output pin.

Input change activity on a port input pin that is enabled for IOC generates an internal device interrupt and the device captures the value of the port and copies it into INTCAP. The interrupt remains active until the INTCAP or GPIO register is read. Writing to these registers does not affect the interrupt. The interrupt condition is cleared after the LSb of the data is clocked out during a read command of GPIO or INTCAP.

The first interrupt event causes the port contents to be copied into the INTCAP register. Subsequent interrupt conditions on the port will not cause an interrupt to occur as long as the interrupt is not cleared by a read of INTCAP or GPIO.

Note: The value in INTCAP can be lost if GPIO is read before INTCAP while another IOC is pending. After reading GPIO, the interrupt will clear and then set due to the pending IOC, causing the INTCAP register to update.

3.6.5 INTERRUPT CONDITIONS

There are two possible configurations that cause interrupts (configured via INTCON):

1. Pins configured for **interrupt-on-pin change** will cause an interrupt to occur if a pin changes to the opposite state. The default state is reset after an interrupt occurs and after clearing the interrupt condition (i.e., after reading GPIO or INTCAP). For example, an interrupt occurs by an input changing from '1' to '0'. The new initial state for the pin is a logic '0' after the interrupt is cleared.
2. Pins configured for **interrupt-on-change from register value** will cause an interrupt to occur if the corresponding input pin differs from the register bit. The interrupt condition will remain as long as the condition exists, regardless if the INTCAP or GPIO is read.

See [Figures 3-8](#) and [3-9](#) for more information on interrupt operations.

FIGURE 3-8: INTERRUPT-ON-PIN CHANGE

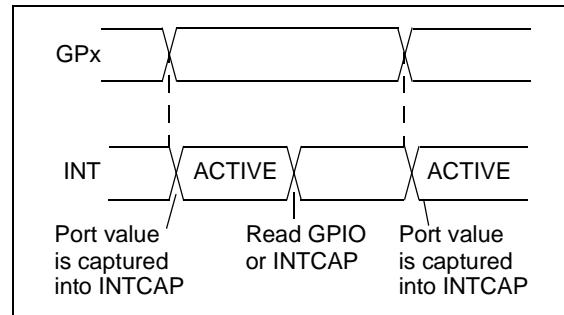
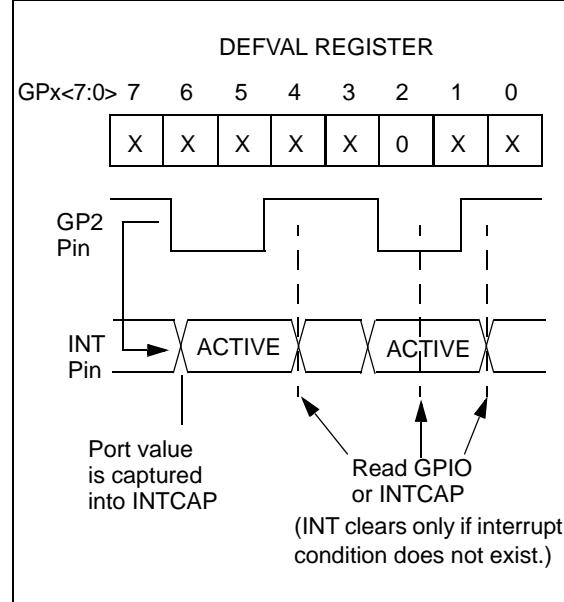


FIGURE 3-9: INTERRUPT-ON-CHANGE FROM REGISTER DEFAULT



MN3209

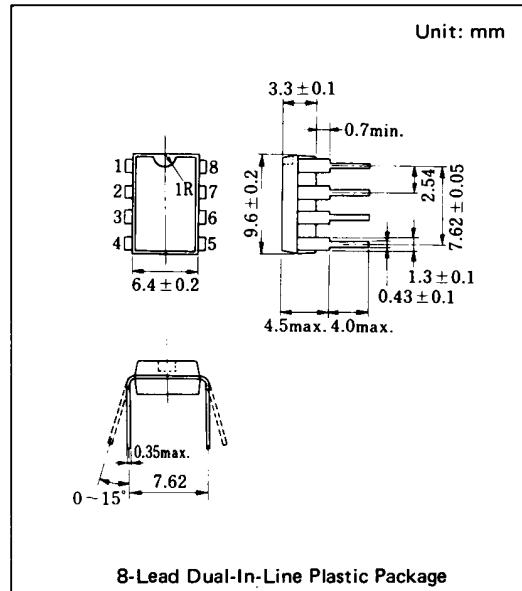
256-STAGE LOW VOLTAGE OPERATION LOW NOISE BBD

■ General description

The MN3209 is a 256-stage low voltage operation ($V_{DD} = 5V$) low noise BBD that provides a signal delay of up to 12.8ms and is particularly suitable as a device for generation of chorus and vibrato effects of audio equipments in low voltage operation portable stereo, radio cassette recorder and electronic musical instruments, etc.

■ Features

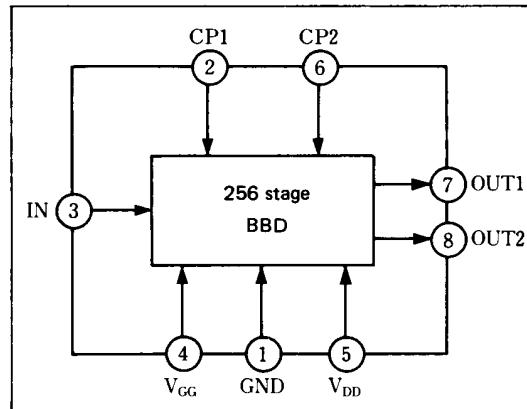
- Variable delay of audio signal: 0.64 ~ 12.8ms.
- Wide supply voltage: 4 ~ 10V.
- No insertion loss: $L_i = 0\text{dB}$ typ.
- Wide dynamic range: S/N = 80dB typ.
- Low distortion: THD = 0.4% typ. ($V_i = 0.25\text{VRms}$).
- Clock frequency range: 10KHz ~ 100KHz.
- N-channel silicon gate process.
- 8-lead dual-in-line plastic package.



■ Applications

- Sound and echo effects of audio equipment such as radio cassette recorder, car radio, portable radio, portable stereo, echo microphone and pre-taped musical accompaniments, etc.
- Sound effect in electronic musical instruments.
- Variable or fixed delay of analog signals.

■ Block Diagram



■ Quick Reference Data

Item	Symbol	Value	Unit
Supply Voltage	V_{DD}, V_{GG}	$+5, \frac{14}{15}V_{DD}$	V
Signal Delay Time	t_D	0.64~12.8	ms
Total Harmonic Distortion	THD	0.4	%
Signal to Noise Ratio	S/N	80	dB

■ Absolute Maximum Ratings (Ta = 25°C)

Item	Symbol	Rating	Unit
Terminal Voltage	V _{DD} , V _{GG} , V _{CPL} , V _I	-0.3~+11	V
Output Voltage	V _O	-0.3~+11	V
Operating Temperature	T _{opr}	-20~+60	°C
Storage Temperature	T _{stg}	-55~+125	°C

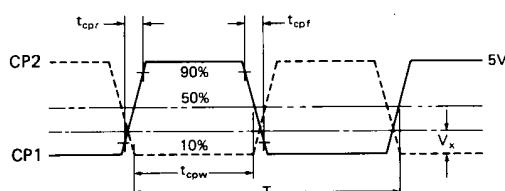
■ Operating Condition (Ta = 25°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Drain Supply Voltage	V _{DD}		+4	+5	+10	V
Gate Supply Voltage	V _{GG}			14/15 V _{DD}		V
Clock Voltage "H" Level	V _{CPLH}			V _{DD}		V
Clock Voltage "L" Level	V _{CPLL}		0		+1	V
Clock Frequency	f _{CP}		10		200	kHz
Clock Pulse Width *1	t _{CPW}				0.5T *2	
Clock Rise Time *1	t _{CPR}				500	ns
Clock Fall Time *1	t _{CPF}				500	ns
Clock Input Capacitance	C _{CP}				200	pF
Clock Cross Point *1	V _X		0		0.3V _{CPLH}	V

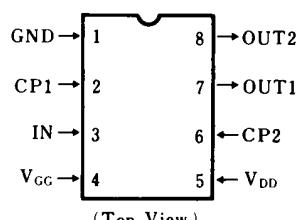
■ Electrical Characteristics (Ta = 25°C, V_{DD} = V_{CPLH} = 5V, V_{CPLL} = 0V, V_{GG} = 14/15 V, R_L = 100kΩ)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Signal Delay Time	t _D		0.64		12.8	ms
Input Signal Frequency	f _i	f _{CP} = 40kHz, Output -3dB down	12			kHz
Input Signal Swing	V _i	THD=2.5%	0.5			Vrms
Insertion Loss	L _i	f _{CP} =40kHz, f _i =1kHz	-4	0	4	dB
Total Harmonic Distortion	THD	f _{CP} =40kHz, f _i =1kHz, V _i =0.25Vrms		0.4	2.5	%
Noise Voltage	V _{no}	f _{CP} = 100kHz Weighted by "A" curve			0.12	mVrms
Signal to Noise Ratio	S/N			80		dB

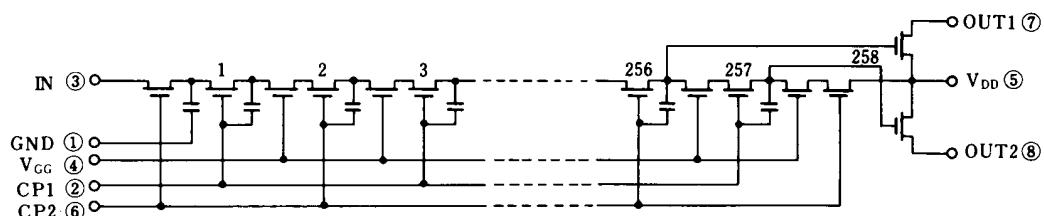
*1 Clock Pulse Waveforms

*2 T = 1/f_{CP} (Clock Period)

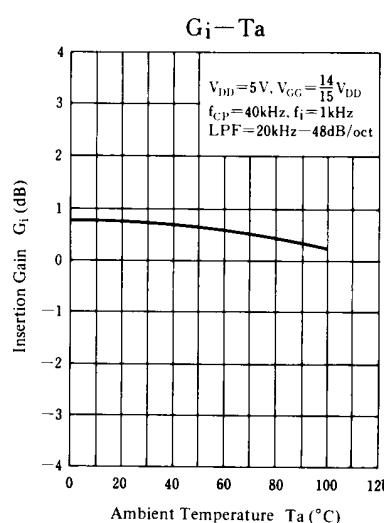
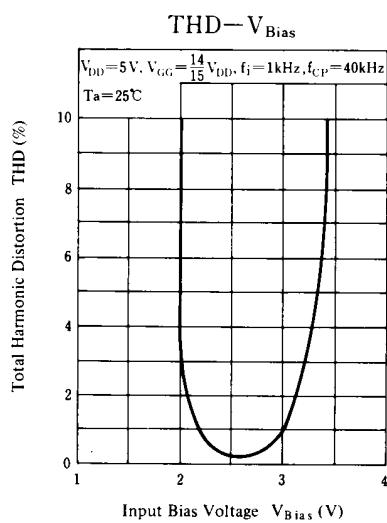
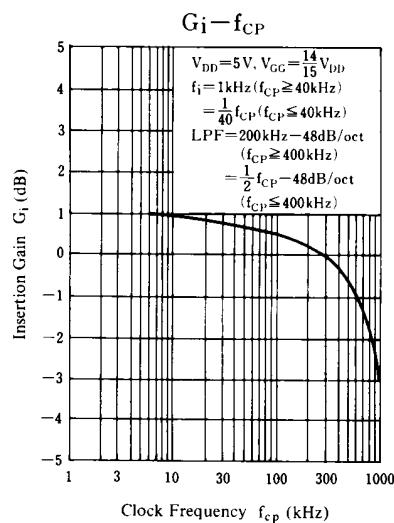
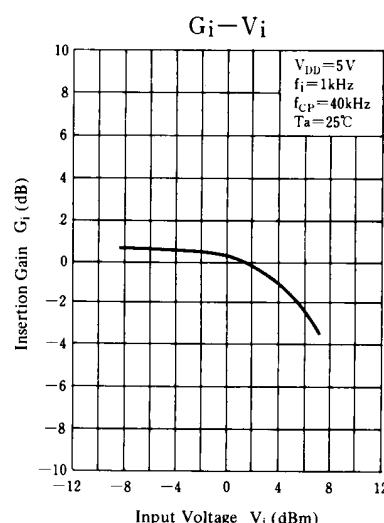
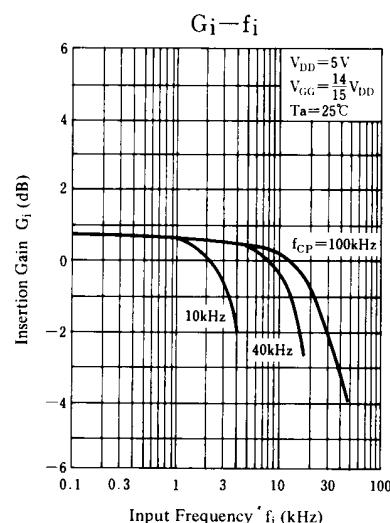
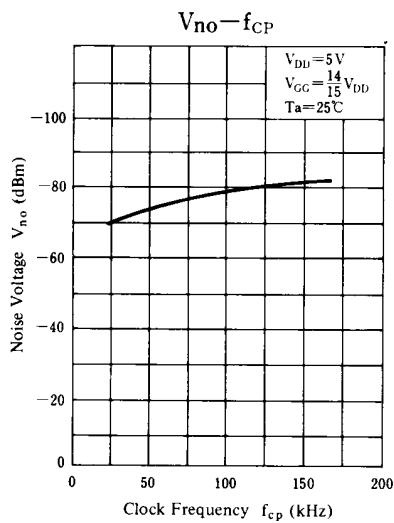
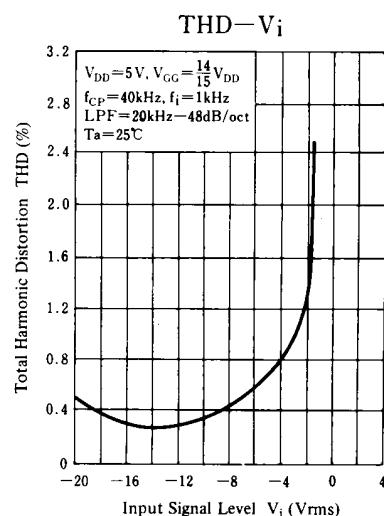
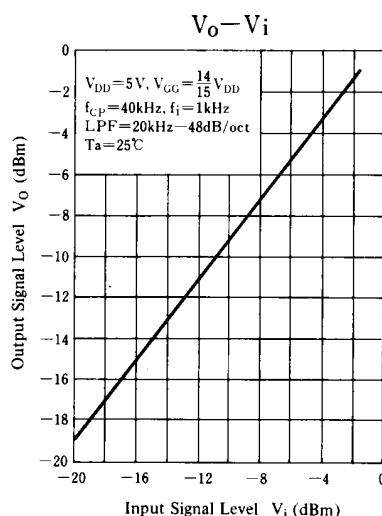
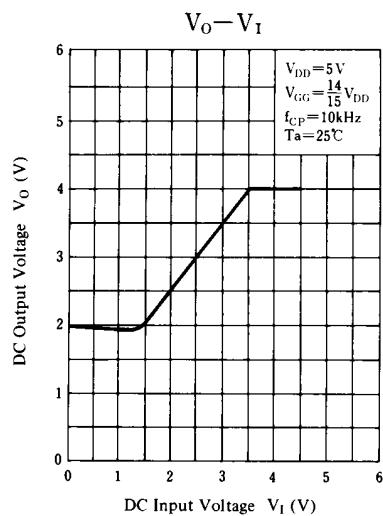
■ Terminal Assignments

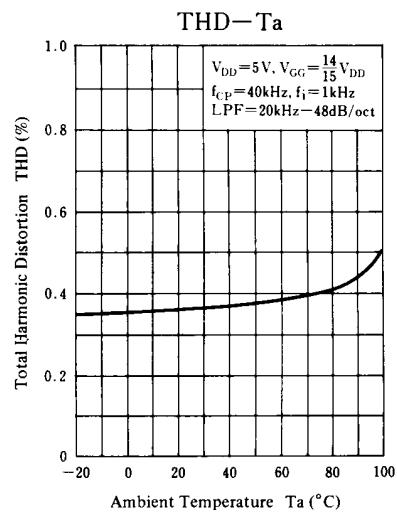
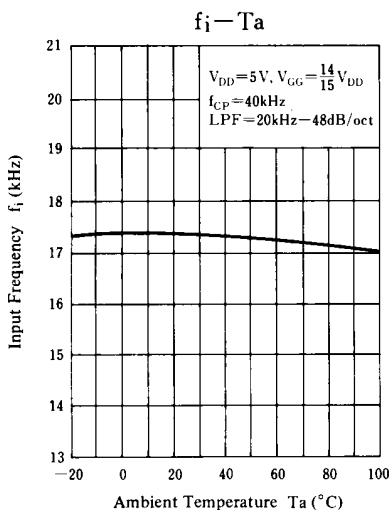
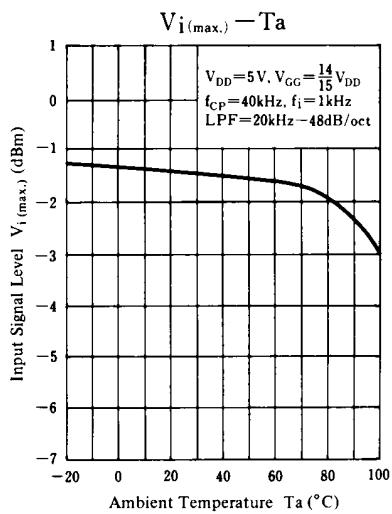


■ Circuit Diagram

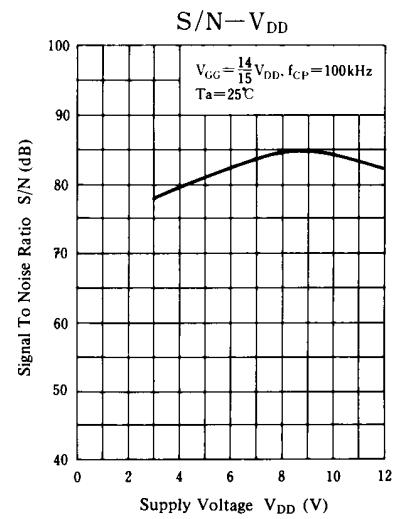
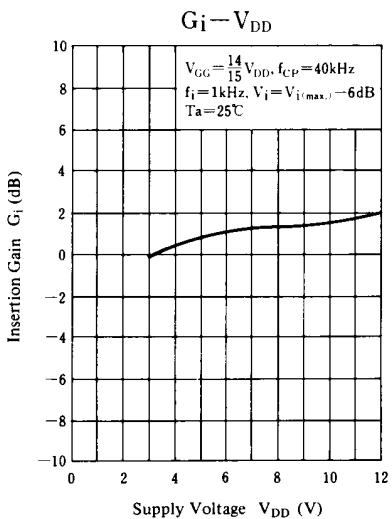
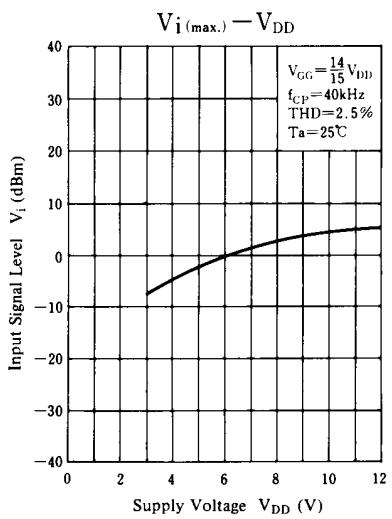
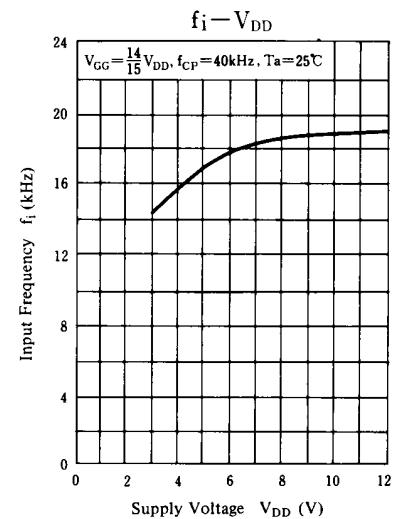
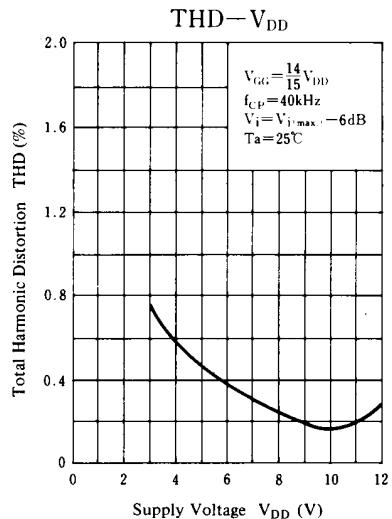
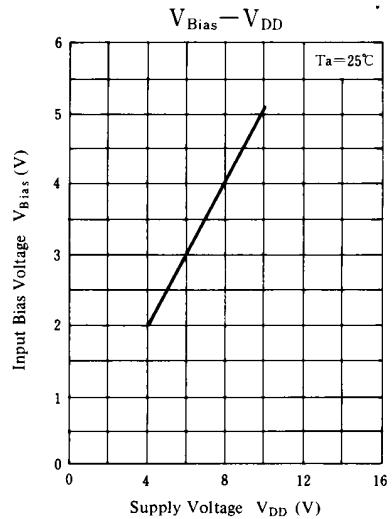


■ Typical Electrical Characteristic Curves

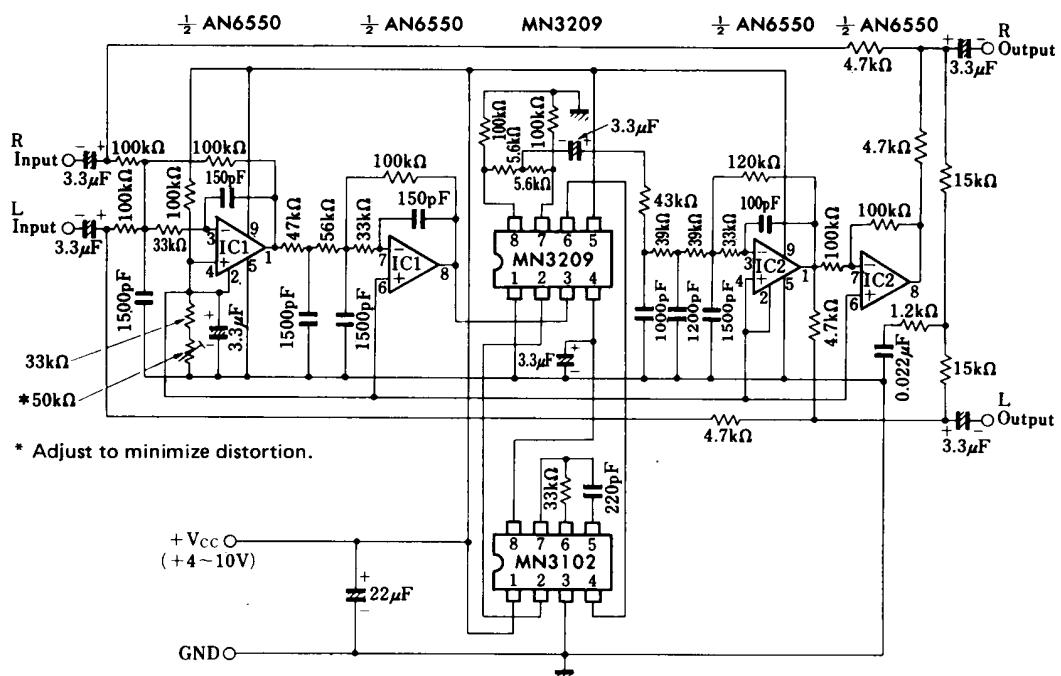




■ Supply Voltage Characteristics



■ Application Circuit



Vibrato and/or Chorus Effect Generation Circuit

DC-Heating
Directly-Heated
Connected in Parallel

KORG

Nutube
6P1
Twin Triode

1. Ratings

Parameter	Symbol	MIN	TYP	MAX	Unit
Operating Temp.	To	-40	-	+85	°C
Storage Temp.	Ts	-50	-	+85	°C
Filament Voltage	Ef #1	0.6	0.7	0.8	V
Grid Voltage	Eg #2	-	-	5.0	V
Anode Voltage	Ea #2	5.0	-	80.0	V
Anode Power Dissipation	Na	-	-	1.7	mW
Insulation Resistance(Anode-Other)	-	3	-	-	MΩ
Insulation Resistance(Grid-Other)	-	3	-	-	MΩ
Color of Illumination	Green				

2. Electrical Characteristics

Parameter	Symbol	Test Condition	MIN	TYP	MAX	Unit	
Filament Current	If	Ef=0.7Vdc #1	Eg=Ea=0V	16.0	17.0	20.0	mAdc
Bias Voltage	Bias		Ia=18.2μA #5#6 Vin=0Vrms #5	1.5	2.5	3.5	Vdc
Voltage Amplification (Vout / Vin)	A		Ia=18.2μA #5#6 Vin=0.245Vrms, 1kHz	3.7	5.0	6.4	-
Resonance Frequency	Fr #7		Ia=18.2μA #5#6 Vin=0Vrms #5	-	5.8	-	kHz
Anode Current	Ig #3		Eg=2.0Vdc #2	-	32.0	-	μAdc
Grid Current	Ia #4		Ea=12.0Vdc	-	6.0	-	μAdc
Amplification Factor	μ		-	14.5	-	-	
Transconductance	gm		-	54	-	-	
Anode Resistance	Ra		-	330	-	kΩ	

- #1 Per each filament
- #2 See Fig.1
- #3 Per each grid
- #4 Per each anode
- #5 See Fig.2
- #6 Ia adjusted by 'Bias'

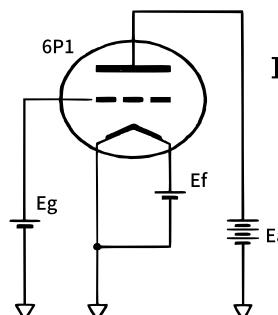


Fig. 1

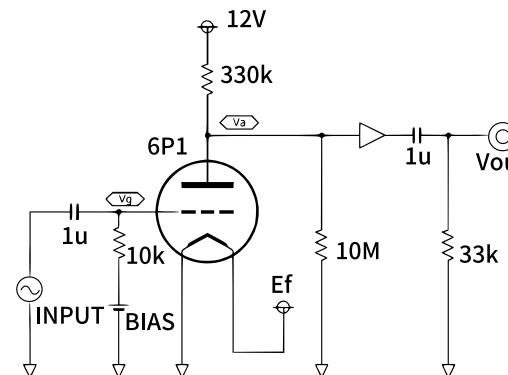


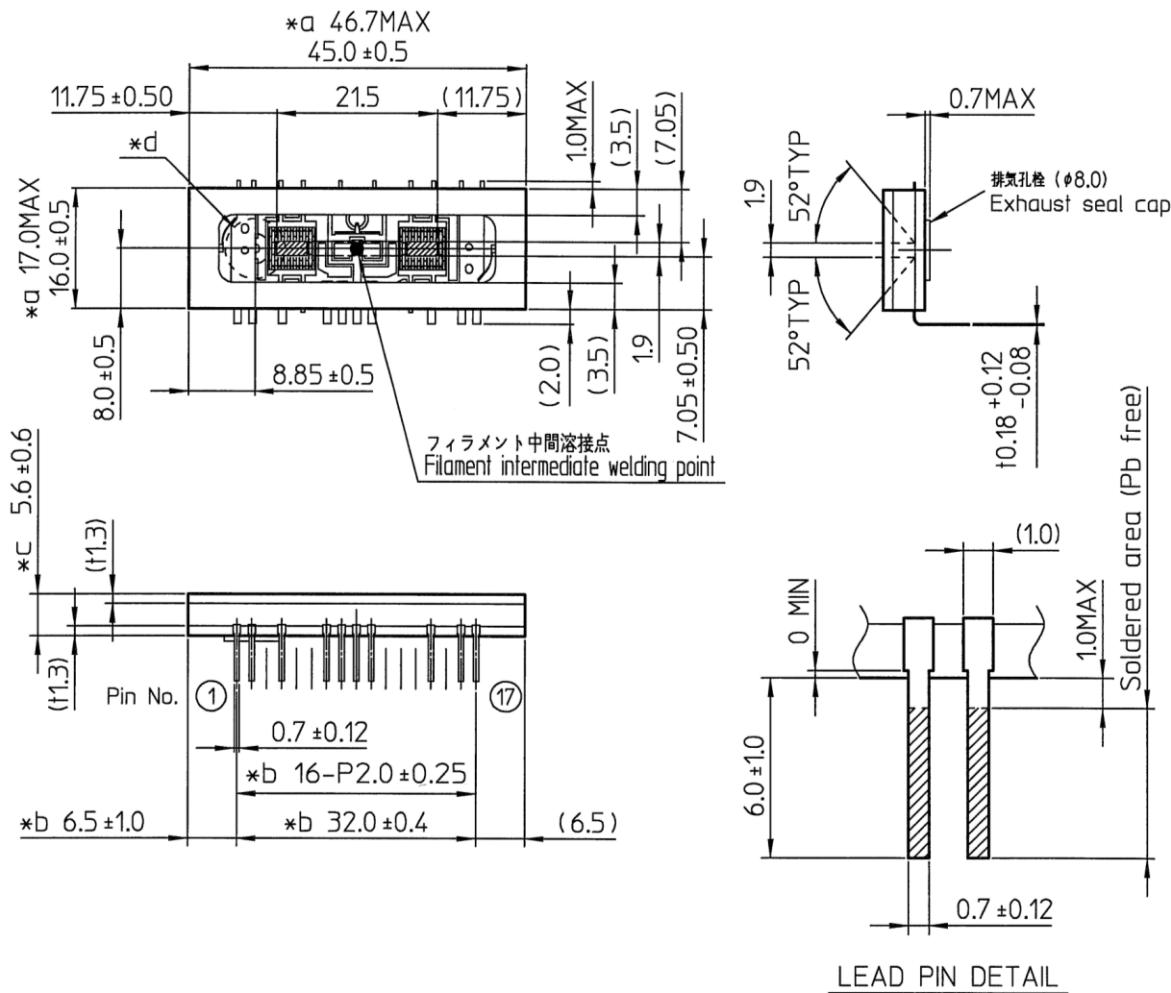
Fig. 2

Capacitances (Typical)

F1-G1	F1-A1	F1-GND	G1-A1	G1-GND	G1-A2	G1-G2	A1-GND	A1-A2
2-4	2-7	2-8	4-7	4-8	4-10	4-14	7-8	7-10
9.1pF	4.3pF	39.2pF	2.5pF	10.9pF	2.3pF	4.3pF	4.5pF	1.4pF

Nutube 6P1 External Dimensions

Unit: mm
(): Reference only
参考寸法



*d フリットのはみ出しを含む寸法とする。

Including any protruding frit glass.

*b 基板底面より3mmの位置の寸法とする。

Within 3mm from the bottom of the glass substrate.

*c 排気孔栓の厚みを含まない。

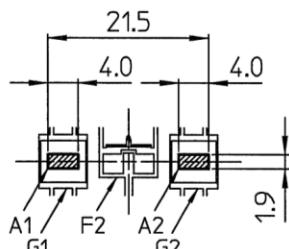
Does not include the thickness of the exhaust seal cap.

*d 排気孔栓は排気孔の中心から半径6mmの範囲に収まっていること。

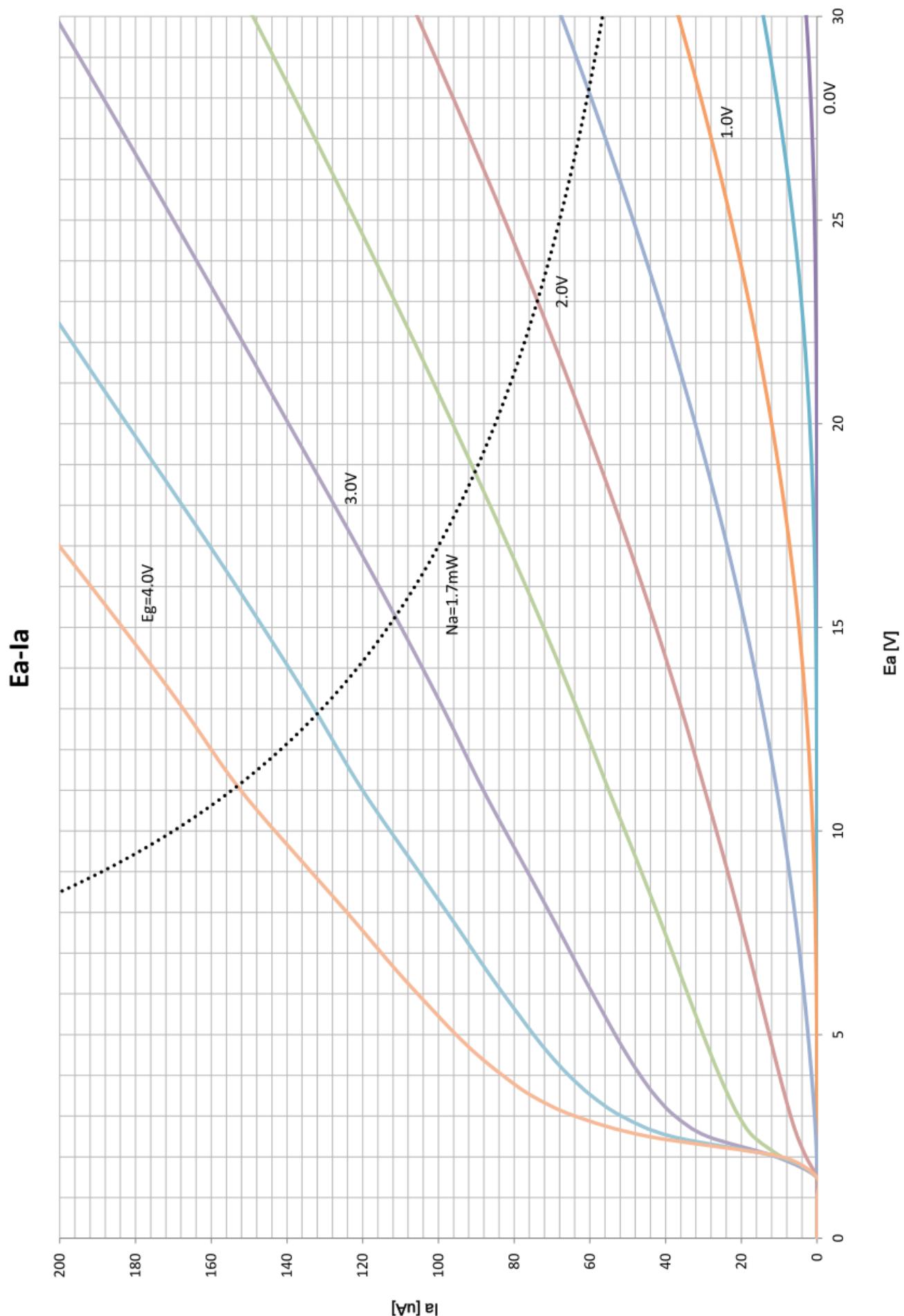
Exhaust seal cap is entirely within a 6mm radius from the center point.

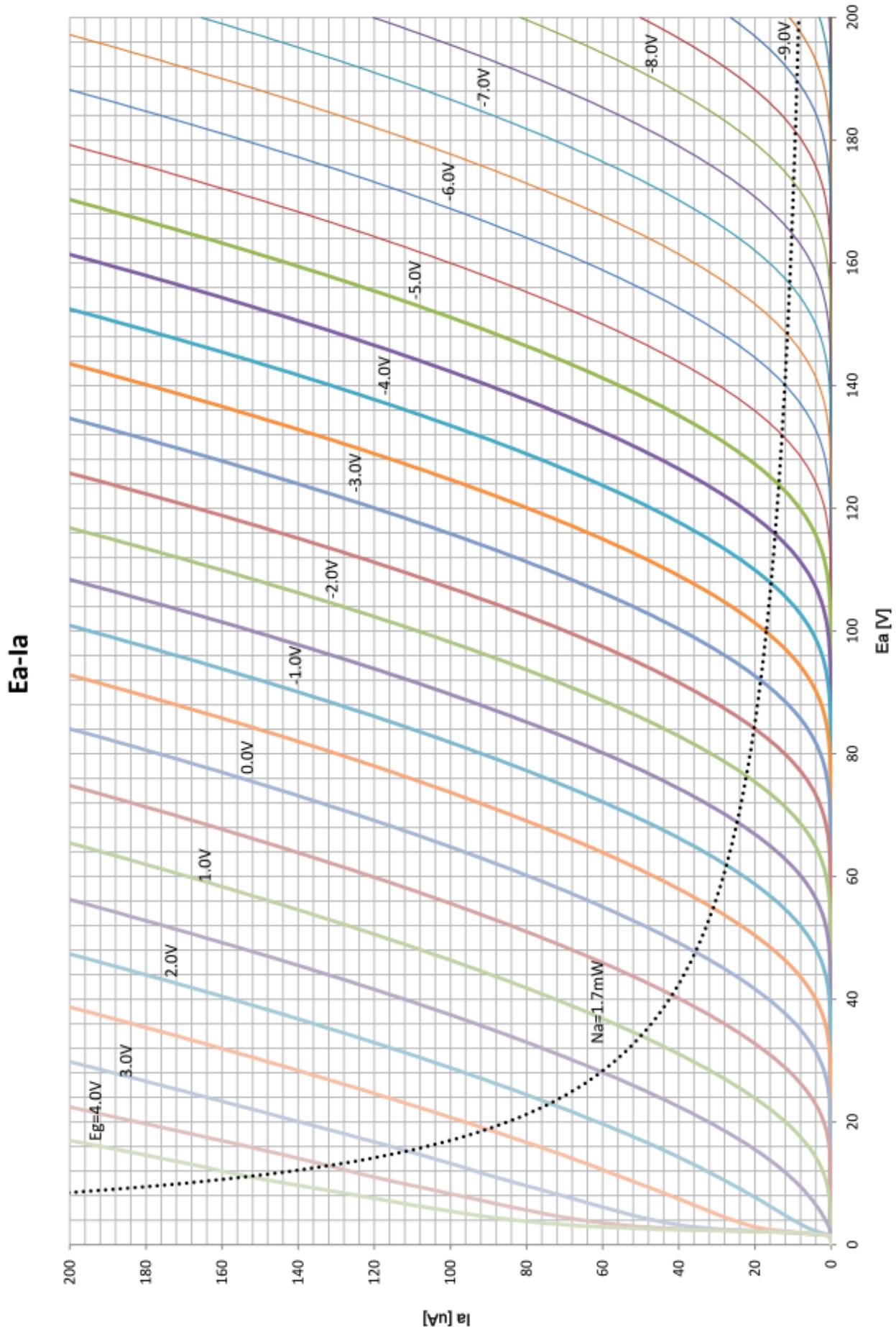
Pin assignment

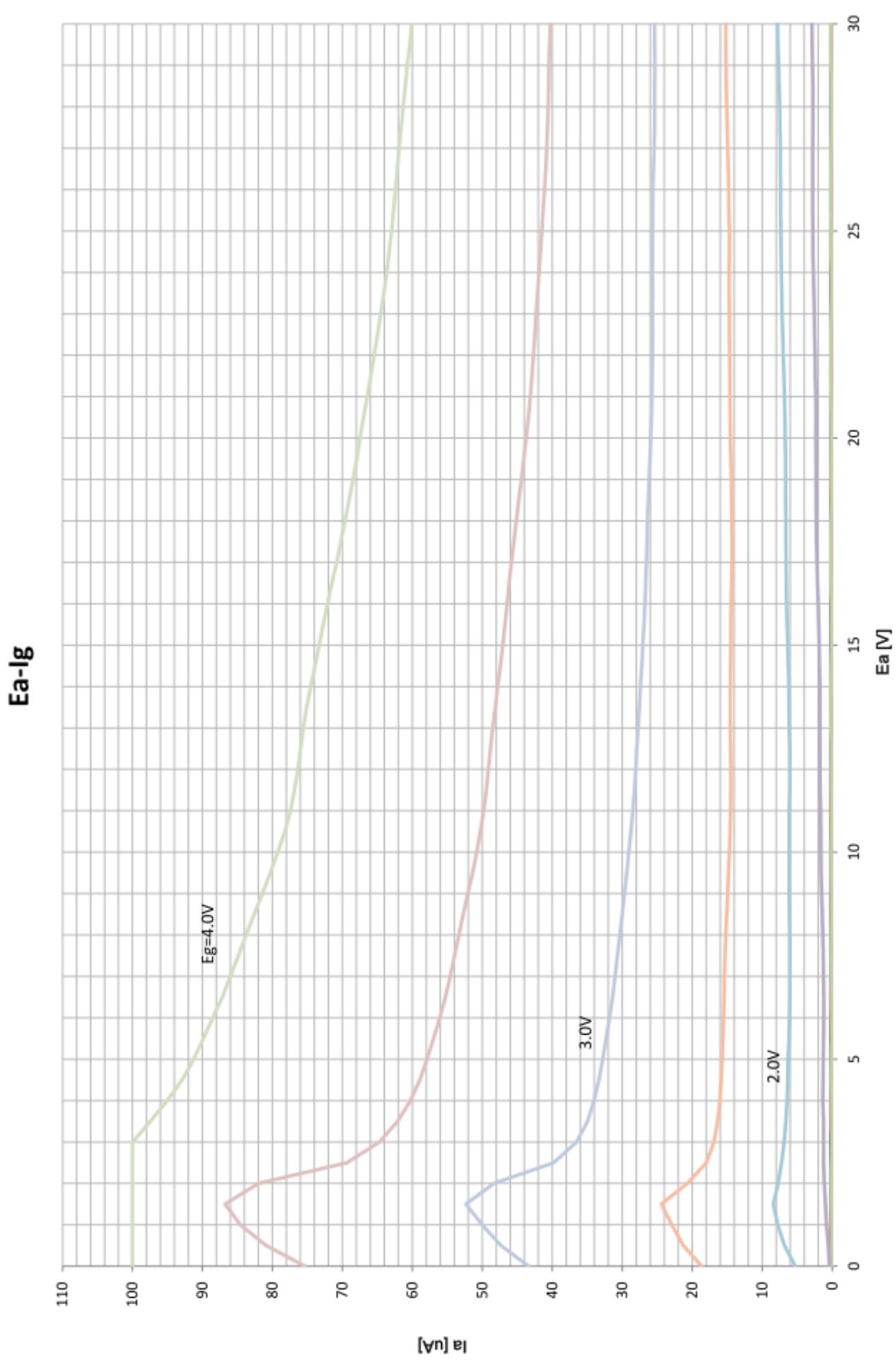
Pin NO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Assignment	F1	F1	NP	G1	NP	NP	A1	GND	F2	A2	NP	NP	NP	G2	NP	F3	F3



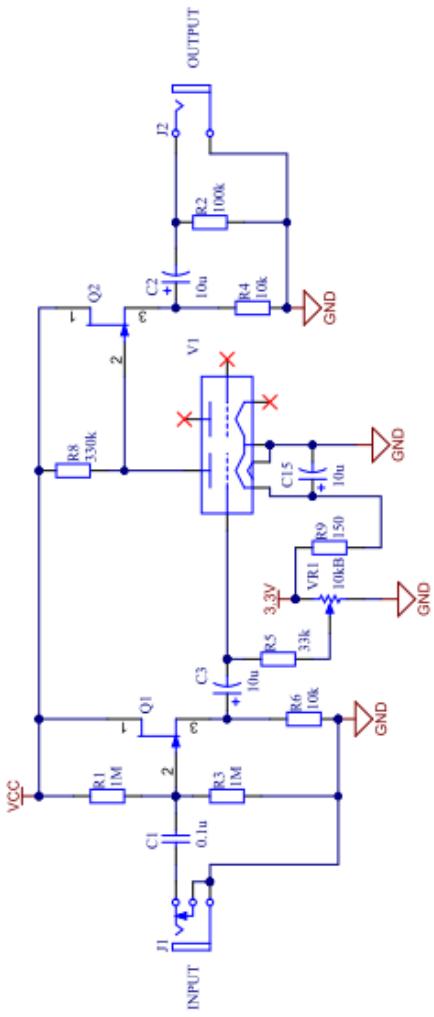
F1: フィラメント 1 (左)/Filament 1 (Left)
 F2: フィラメント 2 (中間)/Filament 2 (Center)
 F3: フィラメント 3 (右)/Filament 3 (Right)
 NP: ノーピン/No Pin
 G1: グリッド 1/Grid 1
 G2: グリッド 2/Grid 2
 A1: アノード 1/Anode 1
 A2: アノード 2/Anode 2







Nutube Basic Circuit



Application notes

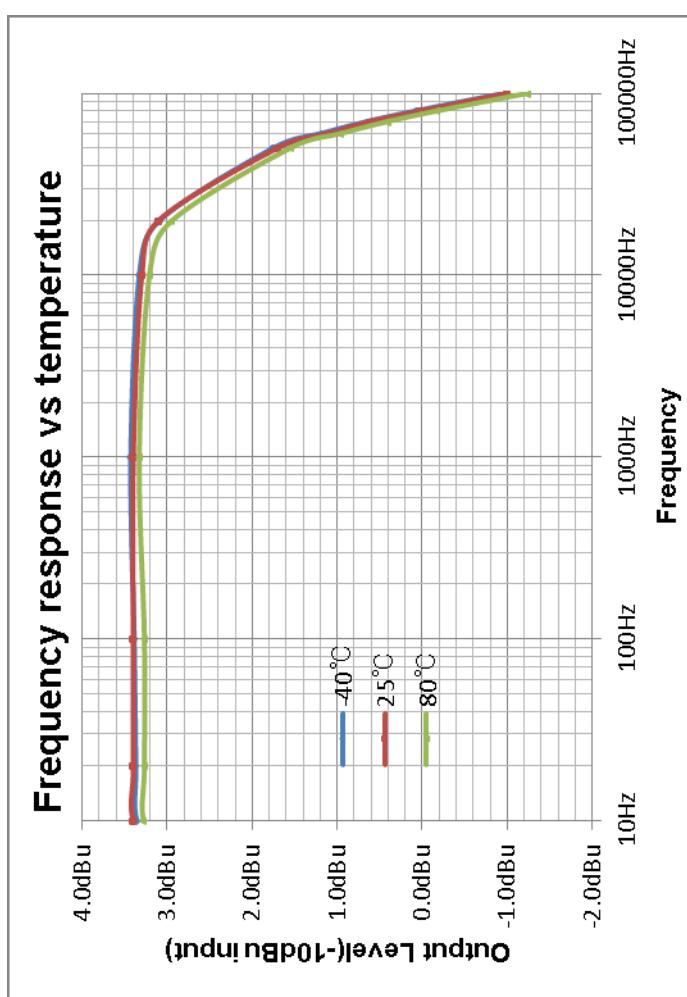
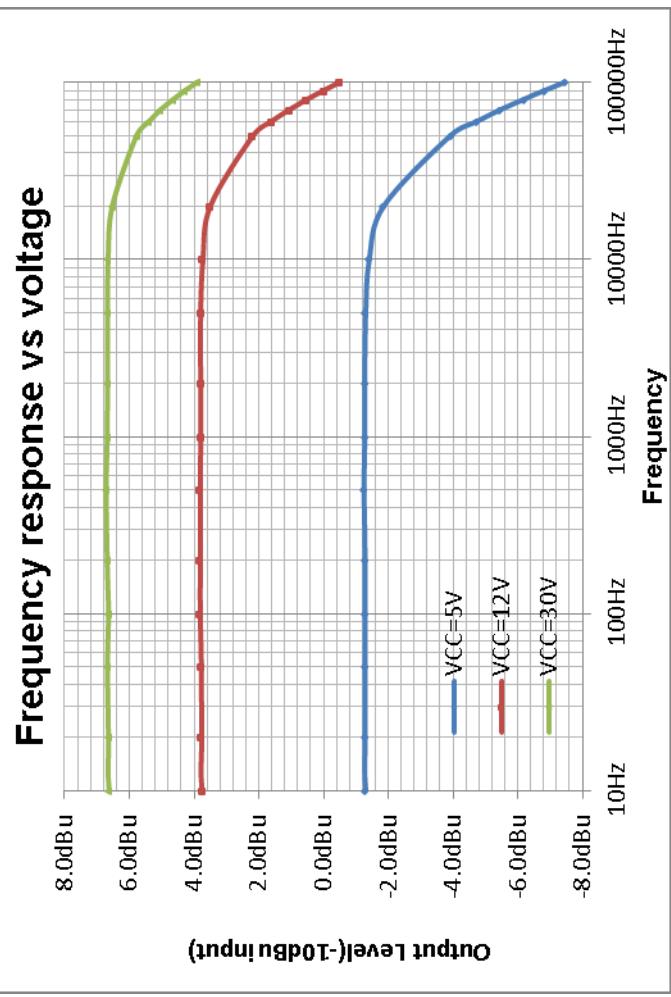
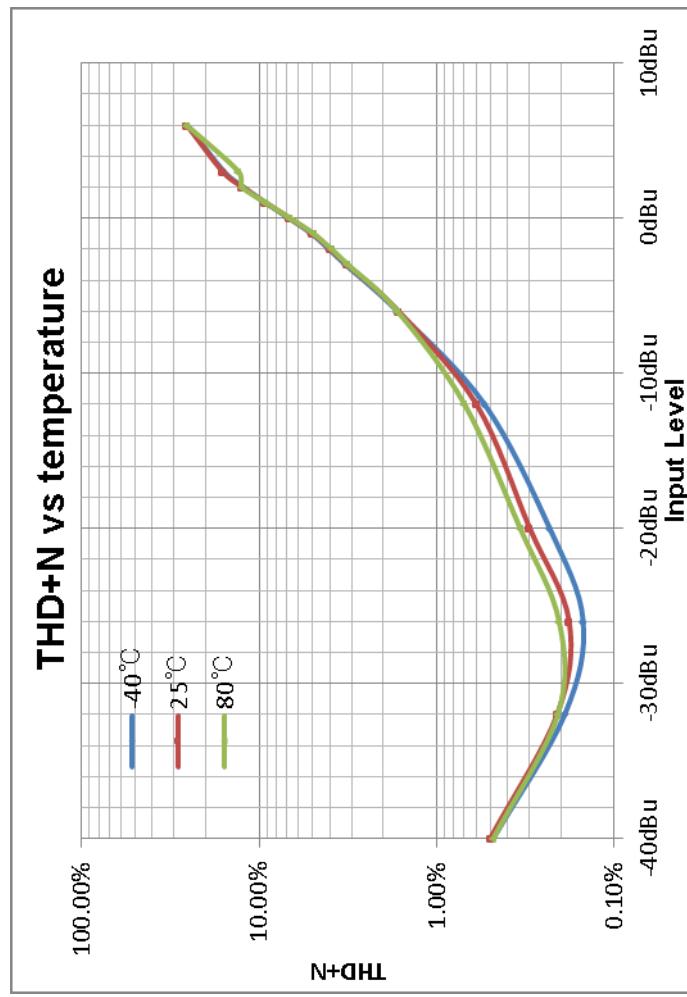
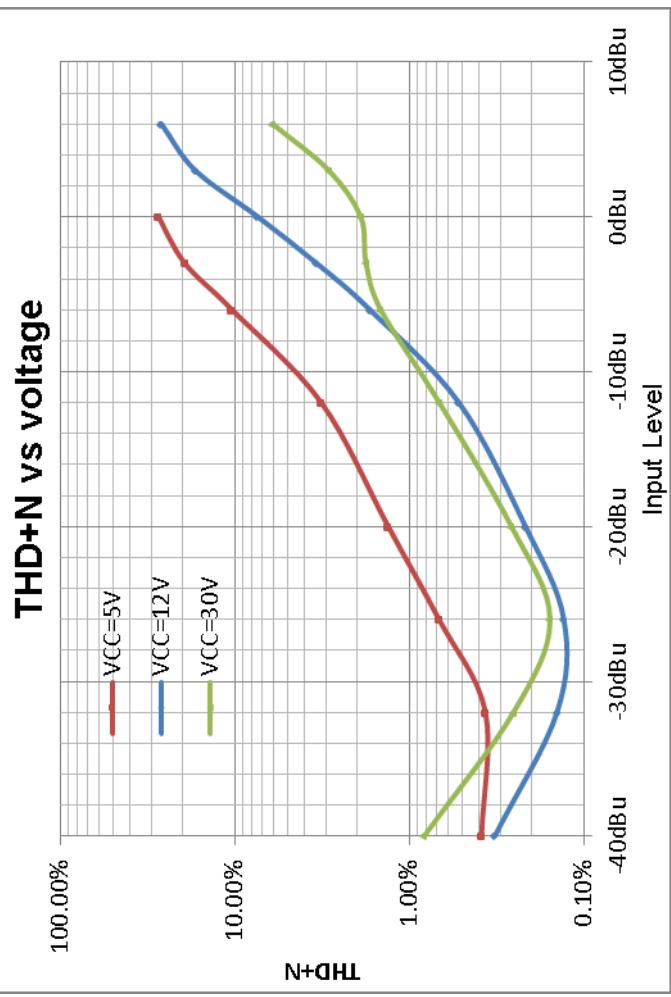
1. C15 reduces residual noise.
When $C15=10\mu F$, residual noise is 9dB less than when without C15.

2. Nutube requires **POSITIVE** grid bias when the VCC voltage is under 40V approximately.
VR1 adjusts the bias voltage.

3. The approximate circuit gains are:

9dB(VCC=5V)
14dB(VCC=12V)
17dB(VCC=30V)

When R8 varies(VCC=12V):
9dB(R8=100k)
13dB(R8=220k)
14dB(R8=330k)



LM13700 Dual Operational Transconductance Amplifiers With Linearizing Diodes and Buffers

1 Features

- g_m Adjustable Over 6 Decades
- Excellent g_m Linearity
- Excellent Matching Between Amplifiers
- Linearizing Diodes for reduced output distortion
- High Impedance Buffers
- High Output Signal-to-Noise Ratio

2 Applications

- Current-Controlled Amplifiers
- Stereo Audio Amplifiers
- Current-Controlled Impedances
- Current-Controlled Filters
- Current-Controlled Oscillators
- Multiplexers
- Timers
- Sample-and-Hold Circuits

3 Description

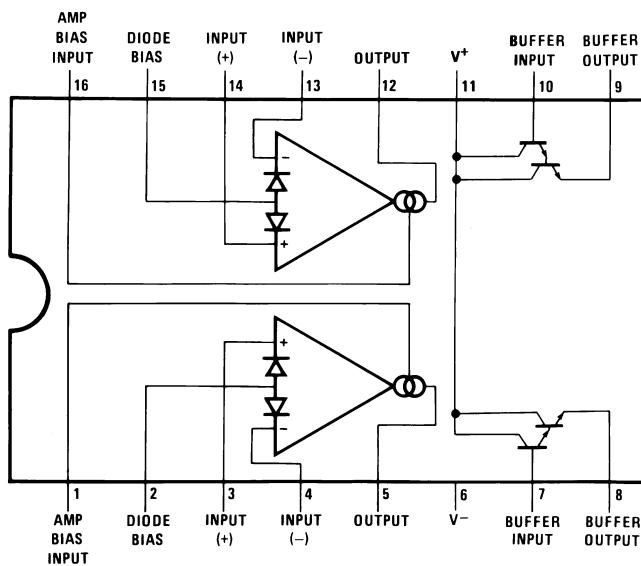
The LM13700 series consists of two current-controlled transconductance amplifiers, each with differential inputs and a push-pull output. The two amplifiers share common supplies but otherwise operate independently. Linearizing diodes are provided at the inputs to reduce distortion and allow higher input levels. The result is a 10-dB signal-to-noise improvement referenced to 0.5 percent THD. High impedance buffers are provided which are especially designed to complement the dynamic range of the amplifiers. The output buffers of the LM13700 differ from those of the LM13600 in that their input bias currents (and thus their output DC levels) are independent of I_{ABC} . This may result in performance superior to that of the LM13600 in audio applications.

Device Information⁽¹⁾

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM13700	SOIC (16)	3.91 mm × 9.90 mm
	PDIP (16)	6.35 mm × 19.304 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Connection Diagram



An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

Table of Contents

1 Features	1	7.4 Device Functional Modes.....	10	
2 Applications	1	8 Application and Implementation	11	
3 Description	1	8.1 Application Information.....	11	
4 Revision History.....	2	8.2 Typical Application	11	
5 Pin Configuration and Functions	3	8.3 System Examples	12	
6 Specifications.....	4	9 Power Supply Recommendations	29	
6.1 Absolute Maximum Ratings	4	10 Layout.....	29	
6.2 Recommended Operating Conditions	4	10.1 Layout Guidelines	29	
6.3 Thermal Information.....	4	10.2 Layout Example	29	
6.4 Electrical Characteristics	5	11 Device and Documentation Support	30	
6.5 Typical Characteristics	6	11.1 Community Resources.....	30	
7 Detailed Description	9	11.2 Trademarks	30	
7.1 Overview	9	11.3 Electrostatic Discharge Caution	30	
7.2 Functional Block Diagram	9	11.4 Glossary	30	
7.3 Feature Description.....	9	12 Mechanical, Packaging, and Orderable	Information	30

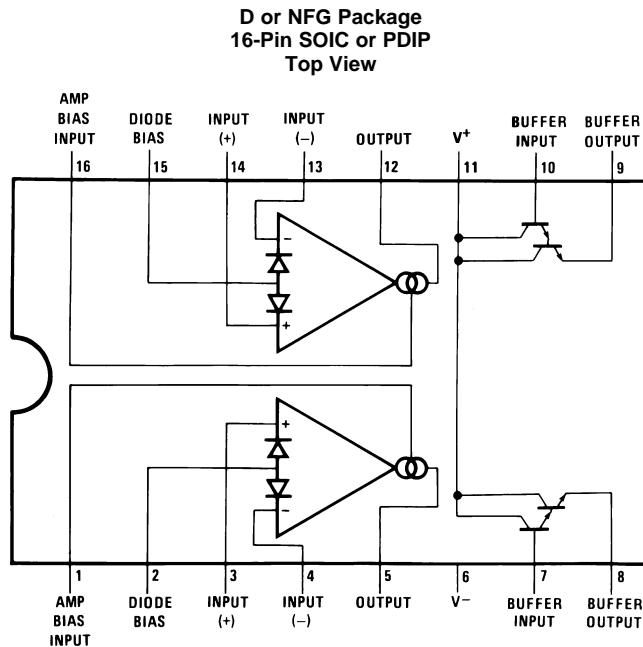
4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision E (March 2013) to Revision F	Page
• Added <i>ESD Ratings</i> table, <i>Feature Description</i> section, <i>Device Functional Modes</i> , <i>Application and Implementation</i> section, <i>Power Supply Recommendations</i> section, <i>Layout</i> section, <i>Device and Documentation Support</i> section, and <i>Mechanical, Packaging, and Orderable Information</i> section.	1
• Removed soldering information in <i>Absolute Maximum Ratings</i> table	4

Changes from Revision D (March 2013) to Revision E	Page
• Changed layout of National Data Sheet to TI format	27

5 Pin Configuration and Functions



Pin Functions

PIN		I/O	DESCRIPTION
NAME	NO.		
Amp bias input	1, 16	A	Current bias input
Buffer input	7, 10	A	Buffer amplifier input
Buffer output	8, 9	A	Buffer amplifier output
Diode bias	2, 15	A	Linearizing diode bias input
Input+	3, 14	A	Positive input
Input-	4, 13	A	Negative input
Output	5, 12	A	Unbuffered output
V ⁺	11	P	Positive power supply
V ⁻	6	P	Negative power supply

6 Specifications

6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)⁽¹⁾

	MIN	MAX	UNIT
Supply voltage		36 V _{DC} or ± 18	V
DC input voltage	+V _S	-V _S	V
Differential input voltage		± 5	V
Diode bias current (I _D)		2	mA
Amplifier bias current (I _{ABC})		2	mA
Buffer output current ⁽²⁾		20	mA
Power dissipation ⁽³⁾ T _A = 25°C – LM13700N		570	mW
Output short circuit duration	Continuous		
Storage temperature, T _{stg}	-65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

(2) Buffer output current should be limited so as to not exceed package dissipation.

(3) For operation at ambient temperatures above 25°C, the device must be derated based on a 150°C maximum junction temperature and a thermal resistance, junction to ambient, as follows: LM13700N, 90°C/W; LM13700M, 110°C/W.

6.2 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

	MIN	MAX	UNIT	
V+ (single-supply configuration)	9.5	32	V	
V+ (dual-supply configuration)	4.75	16	V	
V- (dual-supply configuration)	-16	-4.75	V	
Operating temperature, T _A	LM13700N	0	70	°C

6.3 Thermal Information

THERMAL METRIC ⁽¹⁾	LM13700		UNIT
	D (SOIC)	NFG (PDIP)	
	16 PINS	16 PINS	
R _{θJA} Junction-to-ambient thermal resistance	83.0	43.8	°C/W
R _{θJC(top)} Junction-to-case (top) thermal resistance	44.0	34.9	°C/W
R _{θJB} Junction-to-board thermal resistance	40.5	28.3	°C/W
Ψ _{JT} Junction-to-top characterization parameter	11.5	19.1	°C/W
Ψ _{JB} Junction-to-board characterization parameter	40.2	28.2	°C/W

(1) For more information about traditional and new thermal metrics, see the *Semiconductor and IC Package Thermal Metrics* application report, [SPRA953](#).

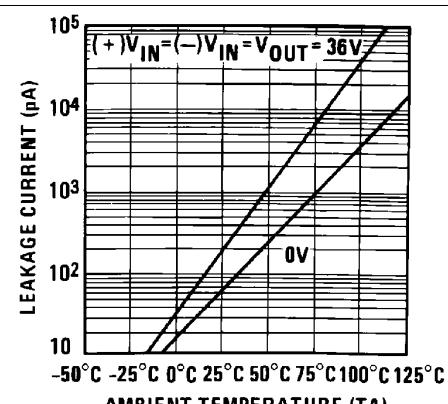
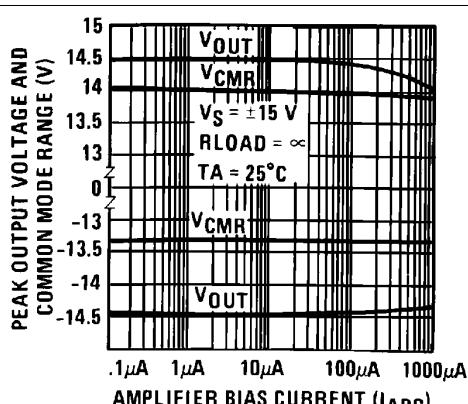
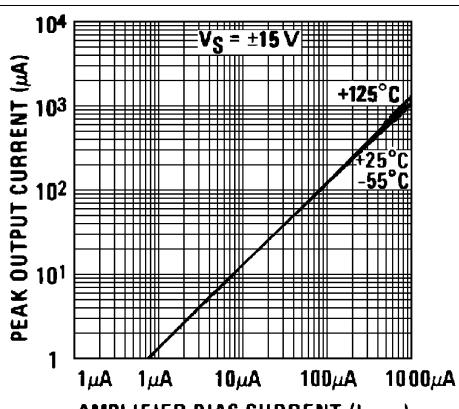
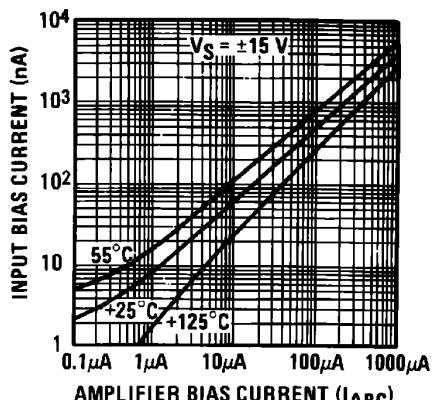
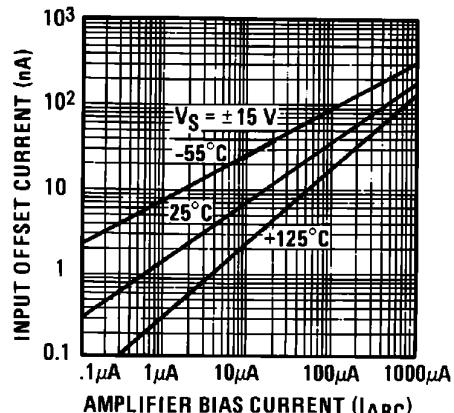
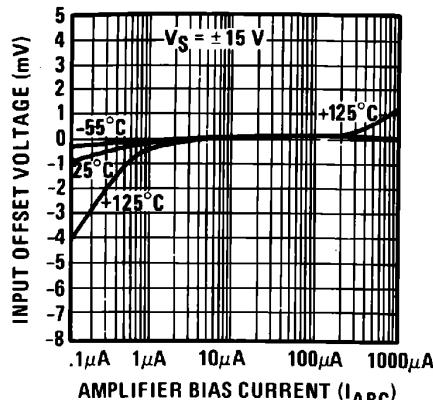
6.4 Electrical Characteristics

These specifications apply for $V_S = \pm 15$ V, $T_A = 25^\circ\text{C}$, amplifier bias current ($I_{ABC} = 500 \mu\text{A}$), pins 2 and 15 open unless otherwise specified. The inputs to the buffers are grounded and outputs are open.

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
Input offset voltage (V_{OS})	Over specified temperature range		0.4	4	mV
	$I_{ABC} = 5 \mu\text{A}$		0.3	4	
V_{OS} including diodes	Diode bias current ($I_D = 500 \mu\text{A}$)		0.5	5	mV
Input offset change	$5 \mu\text{A} \leq I_{ABC} \leq 500 \mu\text{A}$		0.1	3	mV
Input offset current			0.1	0.6	μA
Input bias current			0.4	5	μA
	Over specified temperature range		1	8	
Forward transconductance (g_m)		6700	9600	13000	μS
	Over specified temperature range	5400			
g_m tracking			0.3		dB
Peak output current	$R_L = 0, I_{ABC} = 5 \mu\text{A}$		5		μA
	$R_L = 0, I_{ABC} = 500 \mu\text{A}$	350	500	650	
	$R_L = 0$, Over Specified Temp Range	300			
Supply current	$I_{ABC} = 500 \mu\text{A}$, both channels		2.6		mA
CMRR		80	110		dB
Common-mode range		± 12	± 13.5		V
Crosstalk	Referred to input ⁽¹⁾ $20 \text{ Hz} < f < 20 \text{ kHz}$		100		dB
Differential input current	$I_{ABC} = 0$, input = ± 4 V	0.02	100		nA
Leakage current	$I_{ABC} = 0$ (refer to test circuit)		0.2	100	nA
Input resistance		10	26		k Ω
Open-loop bandwidth			2		MHz
Slew rate	Unity gain compensated		50		V/ μs
Buffer input current	See ⁽¹⁾		0.5	2	μA
Peak buffer output voltage	See ⁽¹⁾	10			V
PEAK OUTPUT VOLTAGE					
Positive	$R_L = \infty, 5 \mu\text{A} \leq I_{ABC} \leq 500 \mu\text{A}$	12	14.2		V
Negative	$R_L = \infty, 5 \mu\text{A} \leq I_{ABC} \leq 500 \mu\text{A}$	-12	-14.4		V
V_{OS} SENSITIVITY					
Positive	$\Delta V_{OS}/\Delta V^+$		20	150	$\mu\text{V/V}$
Negative	$\Delta V_{OS}/\Delta V^-$		20	150	$\mu\text{V/V}$

- (1) These specifications apply for $V_S = \pm 15$ V, $I_{ABC} = 500 \mu\text{A}$, $R_{OUT} = 5\text{-k}\Omega$ connected from the buffer output to $-V_S$ and the input of the buffer is connected to the transconductance amplifier output.

6.5 Typical Characteristics



Typical Characteristics (continued)

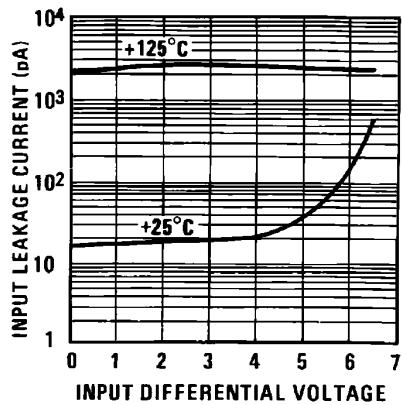


Figure 7. Input Leakage

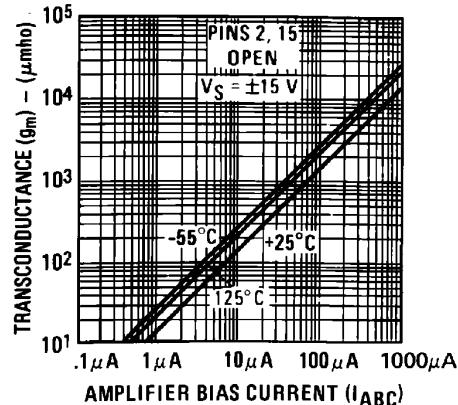


Figure 8. Transconductance

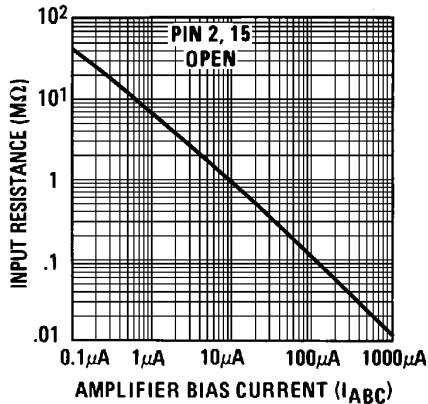


Figure 9. Input Resistance

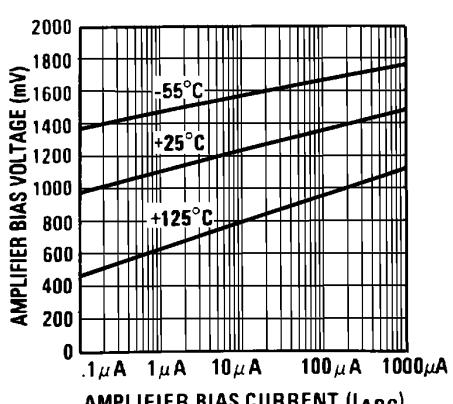


Figure 10. Amplifier Bias Voltage vs. Amplifier Bias Current

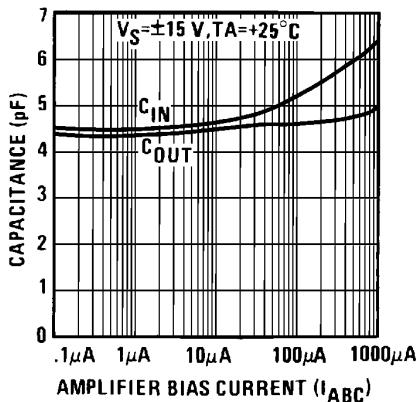


Figure 11. Input and Output Capacitance

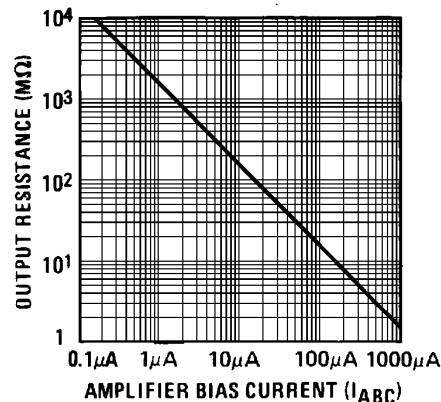


Figure 12. Output Resistance

Typical Characteristics (continued)

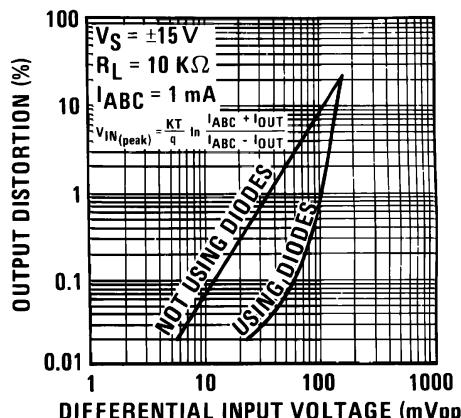


Figure 13. Distortion vs. Differential Input Voltage

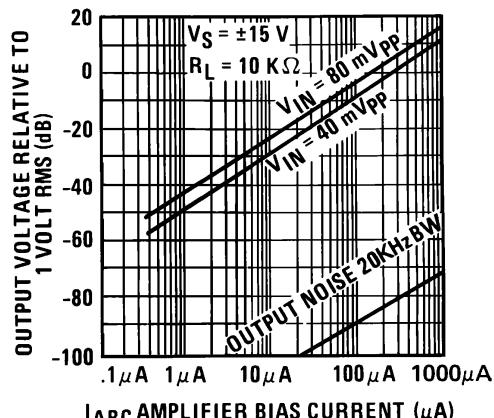


Figure 14. Voltage vs. Amplifier Bias Current

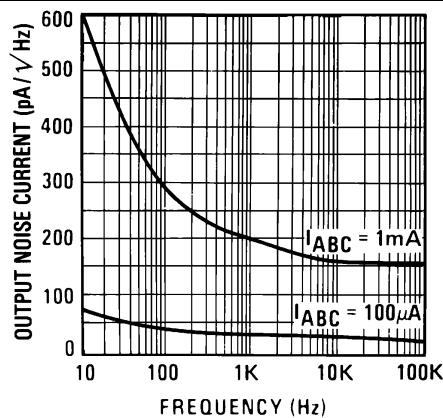


Figure 15. Output Noise vs Frequency

7 Detailed Description

7.1 Overview

The LM13700 is a two channel current controlled differential input transconductance amplifier with additional output buffers. The inputs include linearizing diodes to reduce distortion, and the output current is controlled by a dedicated pin. The outputs can sustain a continuous short to ground.

7.2 Functional Block Diagram

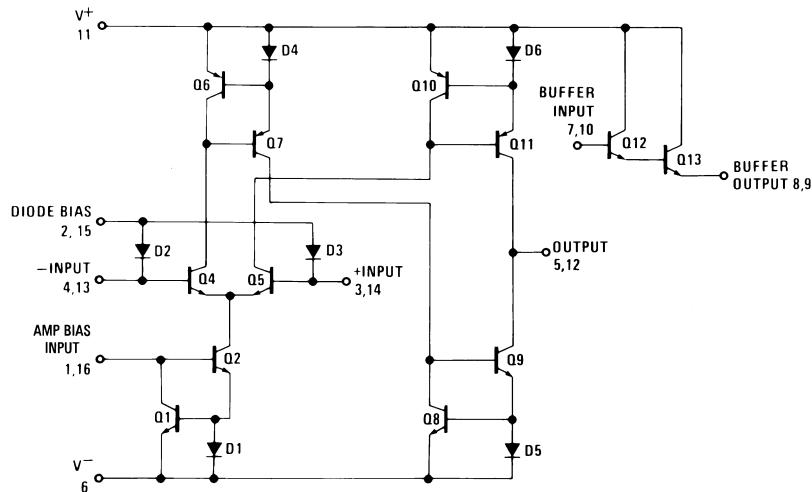


Figure 16. One Operational Transconductance Amplifier

7.3 Feature Description

7.3.1 Circuit Description

The differential transistor pair Q_4 and Q_5 form a transconductance stage in that the ratio of their collector currents is defined by the differential input voltage according to the transfer function:

$$V_{IN} = \frac{kT}{q} \ln \frac{I_5}{I_4} \quad (1)$$

where V_{IN} is the differential input voltage, kT/q is approximately 26 mV at 25°C and I_5 and I_4 are the collector currents of transistors Q_5 and Q_4 respectively. With the exception of Q_{12} and Q_{13} , all transistors and diodes are identical in size. Transistors Q_1 and Q_2 with Diode D_1 form a current mirror which forces the sum of currents I_4 and I_5 to equal I_{ABC} :

$$I_4 + I_5 = I_{ABC} \quad (2)$$

where I_{ABC} is the amplifier bias current applied to the gain pin.

For small differential input voltages the ratio of I_4 and I_5 approaches unity and the Taylor series of the \ln function is approximated as:

$$\begin{aligned} \frac{kT}{q} \ln \frac{I_5}{I_4} &\approx \frac{kT}{q} \frac{I_5 - I_4}{I_4} \\ I_4 &\approx I_5 \approx \frac{I_{ABC}}{2} \end{aligned} \quad (3)$$

$$V_{IN} \left[\frac{I_{ABC}^2}{2kT} \right] = I_5 - I_4 \quad (4)$$

Feature Description (continued)

Collector currents I_4 and I_5 are not very useful by themselves and it is necessary to subtract one current from the other. The remaining transistors and diodes form three current mirrors that produce an output current equal to I_5 minus I_4 thus:

$$V_{IN} \left[\frac{I_{ABC}q}{2kT} \right] = I_{OUT} \quad (5)$$

The term in brackets is then the transconductance of the amplifier and is proportional to I_{ABC} .

7.3.2 Linearizing Diodes

For differential voltages greater than a few millivolts, [Equation 3](#) becomes less valid and the transconductance becomes increasingly nonlinear. [Figure 19](#) demonstrates how the internal diodes can linearize the transfer function of the amplifier. For convenience assume the diodes are biased with current sources and the input signal is in the form of current I_S . Since the sum of I_4 and I_5 is I_{ABC} and the difference is I_{OUT} , currents I_4 and I_5 is written as follows:

$$I_4 = \frac{I_{ABC}}{2} - \frac{I_{OUT}}{2}, \quad I_5 = \frac{I_{ABC}}{2} + \frac{I_{OUT}}{2} \quad (6)$$

Since the diodes and the input transistors have identical geometries and are subject to similar voltages and temperatures, the following is true:

$$\begin{aligned} \frac{kT}{q} \ln \frac{\frac{I_D}{2} + I_S}{\frac{I_D}{2} - I_S} &= \frac{kT}{q} \ln \frac{\frac{I_{ABC}}{2} + \frac{I_{OUT}}{2}}{\frac{I_{ABC}}{2} - \frac{I_{OUT}}{2}} \\ \therefore I_{OUT} &= I_S \left(\frac{2I_{ABC}}{I_D} \right) \text{ for } |I_S| < \frac{I_D}{2} \end{aligned} \quad (7)$$

Notice that in deriving [Equation 7](#) no approximations have been made and there are no temperature-dependent terms. The limitations are that the signal current not exceed $I_D / 2$ and that the diodes be biased with currents. In practice, replacing the current sources with resistors will generate insignificant errors.

7.4 Device Functional Modes

Use in single ended or dual supply systems requires minimal changes. The outputs can support a sustained short to ground. Note that use of the LM13700 in ± 5 V supply systems requires will reduce signal dynamic range; this is due to the PNP transistors having a higher V_{BE} than the NPN transistors.

7.4.1 Output Buffers

Each channel includes a separate output buffer which consists of a Darlington pair transistor that can drive up to 20mA.

8 Application and Implementation

NOTE

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

8.1 Application Information

An OTA is a versatile building block analog component that can be considered an ideal transistor. The LM13700 can be used in a wide variety of applications, from voltage-controlled amplifiers and filters to VCOs. The 2 well-matched, independent channels make the LDC13700 well suited for stereo audio applications.

8.2 Typical Application

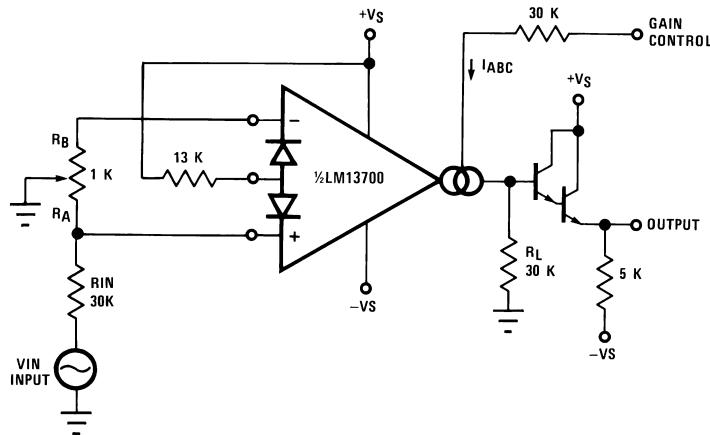


Figure 17. Voltage Controlled Amplifier

8.2.1 Design Requirements

For this example application, the system requirements provide a volume control for a 1 V_P input signal with a THD < 0.1% using ±15 V supplies. The volume control varies between -13 V and 15 V and needs to provide an adjustable gain range of >30dB.

8.2.2 Detailed Design Procedure

Using the linearizing diodes is recommended for most applications, as they greatly reduce the output distortion. It is required that the diode bias current, I_D be greater than twice the input current, I_S . As the input voltage has a DC level of 0 V, the Diode Bias input pins are 1 diode drop above 0 V, which is +0.7 V. Tying the bias to the clean V+ supply, results in a voltage drop of 14.3 V across R_D . Using the recommended 1mA for I_D is appropriate here, and with VS=+15 V, the voltage drop is 14.3 V, and so using the standard value of 13-kΩ is acceptable and will provide the desired gain control.

To obtain the <0.1% THD requirement, the differential input voltage must be <60mVpp when the linearizing diodes are used. The input divider on the input will reduce the 1 V_P input to 33mV_{PP}, which is within the desired spec.

Next, set I_{BIAS} . The Bias Input pins (pins 1 or 16), are 2 diode drops above the negative supply, and therefore $V_{BIAS} = 2(V_{BE}) + V^-$, which for this application is -13.6 V. To set I_{BIAS} to 1ma when $V_C = 15$ V requires a 28.6-kΩ; 30-kΩ is a standard value and is used for this application. The gain will be linear with the applied voltage.

Typical Application (continued)

8.2.3 Application Curve

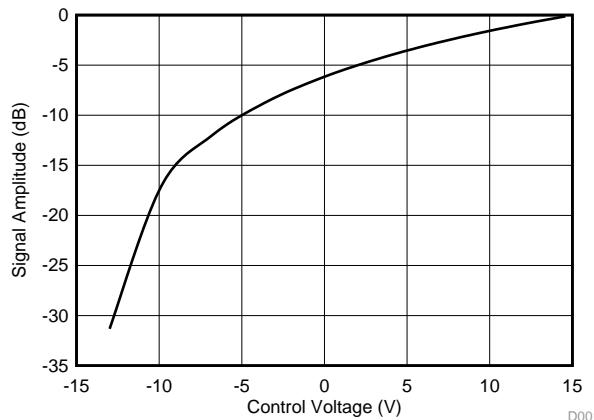


Figure 18. Signal Amplitude vs Control Voltage

8.3 System Examples

8.3.1 Voltage-Controlled Amplifiers

Figure 20 shows how the linearizing diodes is used in a voltage-controlled amplifier. To understand the input biasing, it is best to consider the 13-k Ω resistor as a current source and use a Thevenin equivalent circuit as shown in Figure 21. This circuit is similar to Figure 19 and operates the same. The potentiometer in Figure 20 is adjusted to minimize the effects of the control signal at the output.

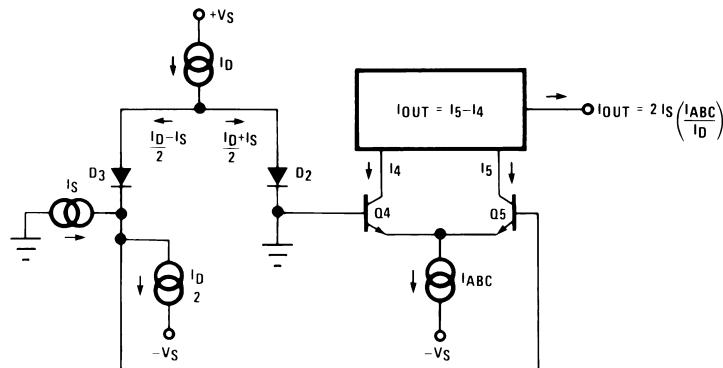


Figure 19. Linearizing Diodes

For optimum signal-to-noise performance, I_{ABC} should be as large as possible as shown by the Output Voltage vs Amplifier Bias Current graph. Larger amplitudes of input signal also improve the S/N ratio. The linearizing diodes help here by allowing larger input signals for the same output distortion as shown by the Distortion vs. Differential Input Voltage graph. S/N may be optimized by adjusting the magnitude of the input signal via R_{IN} (Figure 20) until the output distortion is below the desired level. The output voltage swing can then be set at any level by selecting R_L .

Although the noise contribution of the linearizing diodes is negligible relative to the contribution of the amplifier's internal transistors, I_D should be as large as possible. This minimizes the dynamic junction resistance of the diodes (r_e) and maximizes their linearizing action when balanced against R_{IN} . A value of 1 mA is recommended for I_D unless the specific application demands otherwise.

System Examples (continued)

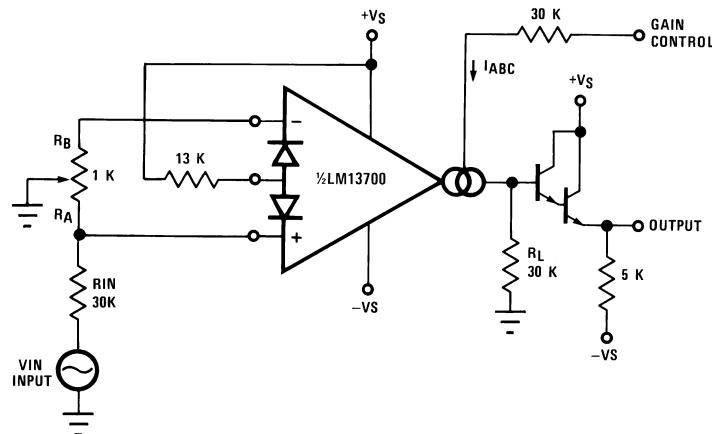


Figure 20. Voltage-Controlled Amplifier

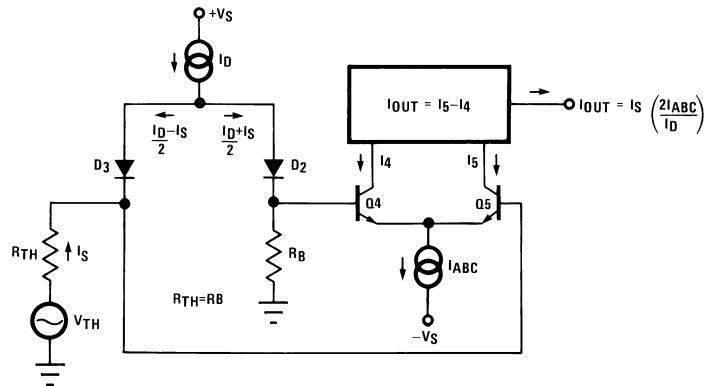


Figure 21. Equivalent VCA Input Circuit

8.3.2 Stereo Volume Control

The circuit of [Figure 22](#) uses the excellent matching of the two LM13700 amplifiers to provide a Stereo Volume Control with a typical channel-to-channel gain tracking of 0.3 dB. R_P is provided to minimize the output offset voltage and may be replaced with two 510Ω resistors in AC-coupled applications. For the component values given, amplifier gain is derived for [Figure 20](#) as being:

$$\frac{V_O}{V_{IN}} = 940 \times I_{ABC} \quad (8)$$

If V_C is derived from a second signal source then the circuit becomes an amplitude modulator or two-quadrant multiplier as shown in [Figure 23](#), where:

$$I_O = \frac{-2I_S}{I_D} (I_{ABC}) = \frac{-2I_S}{I_D} \frac{V_{IN2}}{R_C} - \frac{2I_S}{I_D} \frac{(V^- + 1.4V)}{R_C} \quad (9)$$

The constant term in the above equation may be cancelled by feeding $I_S \times I_D R_C / 2(V^- + 1.4V)$ into I_O . The circuit of [Figure 24](#) adds R_M to provide this current, resulting in a four-quadrant multiplier where R_C is trimmed such that $V_O = 0V$ for $V_{IN2} = 0V$. R_M also serves as the load resistor for I_O .

System Examples (continued)

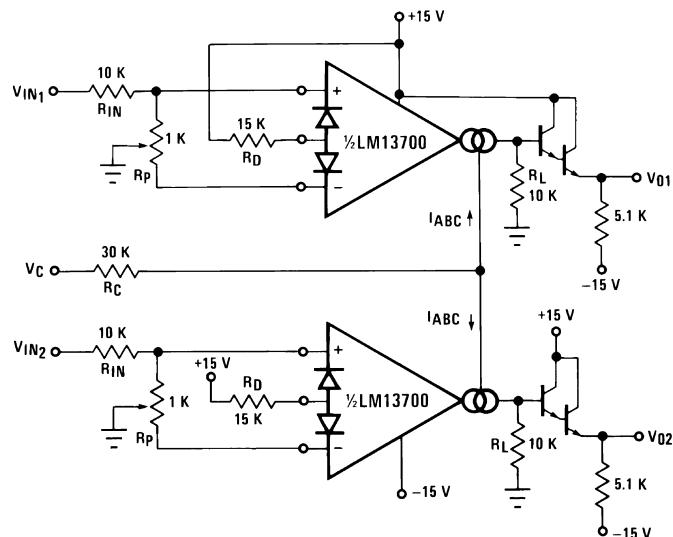


Figure 22. Stereo Volume Control

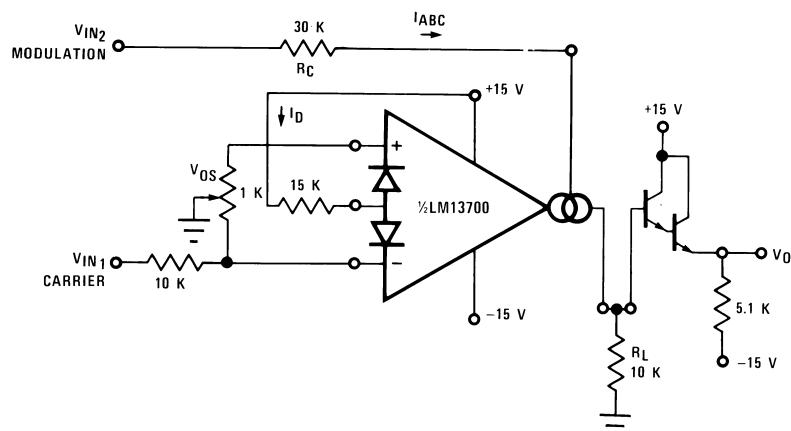


Figure 23. Amplitude Modulator

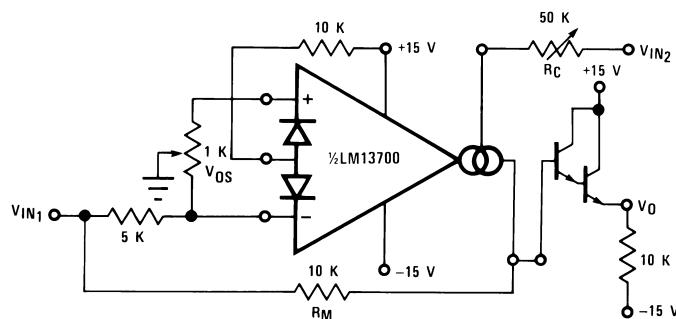


Figure 24. Four-Quadrant Multiplier

System Examples (continued)

Noting that the gain of the LM13700 amplifier of Figure 21 may be controlled by varying the linearizing diode current I_D as well as by varying I_{ABC} , Figure 25 shows an AGC Amplifier using this approach. As V_O reaches a high enough amplitude (3 V_{BE}) to turn on the Darlington transistors and the linearizing diodes, the increase in I_D reduces the amplifier gain so as to hold V_O at that level.

8.3.3 Voltage-Controlled Resistors

An Operational Transconductance Amplifier (OTA) may be used to implement a Voltage Controlled Resistor as shown in Figure 26. A signal voltage applied at R_X generates a V_{IN} to the LM13700 which is then multiplied by the g_m of the amplifier to produce an output current, thus:

$$R_X = \frac{R + R_A}{g_m R_A} \quad (10)$$

where $g_m \approx 19.2 I_{ABC}$ at 25°C. Note that the attenuation of V_O by R and R_A is necessary to maintain V_{IN} within the linear range of the LM13700 input.

Figure 27 shows a similar VCR where the linearizing diodes are added, essentially improving the noise performance of the resistor. A floating VCR is shown in Figure 28, where each “end” of the “resistor” may be at any voltage within the output voltage range of the LM13700.

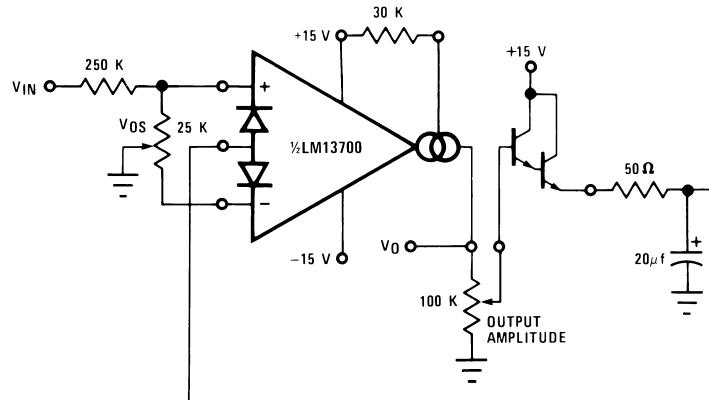


Figure 25. AGC Amplifier

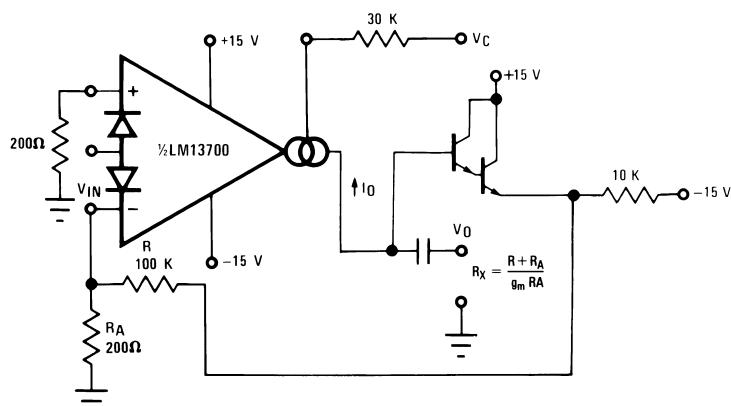


Figure 26. Voltage-Controlled Resistor, Single-Ended

System Examples (continued)

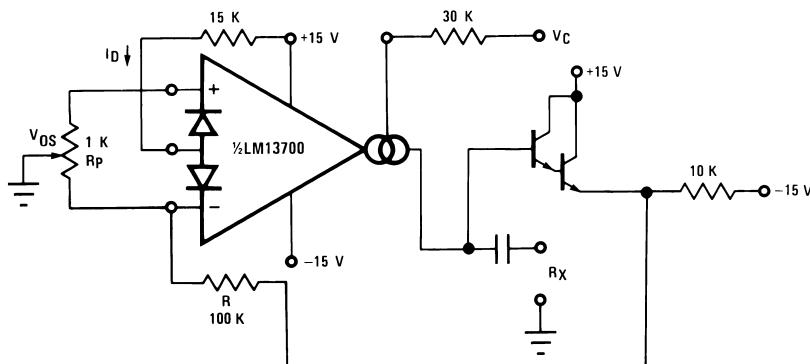


Figure 27. Voltage-Controlled Resistor with Linearizing Diodes

8.3.4 Voltage-Controlled Filters

OTA's are extremely useful for implementing voltage controlled filters, with the LM13700 having the advantage that the required buffers are included on the I.C. The VC Lo-Pass Filter of Figure 29 performs as a unity-gain buffer amplifier at frequencies below cut-off, with the cut-off frequency being the point at which X_C/g_m equals the closed-loop gain of (R/R_A) . At frequencies above cut-off the circuit provides a single RC roll-off (6 dB per octave) of the input signal amplitude with a -3 dB point defined by the given equation, where g_m is again $19.2 \times I_{ABC}$ at room temperature. Figure 30 shows a VC High-Pass Filter which operates in much the same manner, providing a single RC roll-off below the defined cut-off frequency.

Additional amplifiers may be used to implement higher order filters as demonstrated by the two-pole Butterworth Lo-Pass Filter of Figure 31 and the state variable filter of Figure 32. Due to the excellent g_m tracking of the two amplifiers, these filters perform well over several decades of frequency.

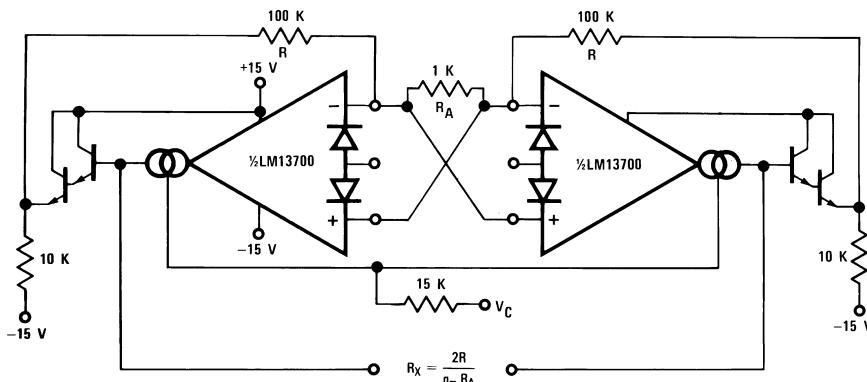


Figure 28. Floating Voltage-Controlled Resistor

System Examples (continued)

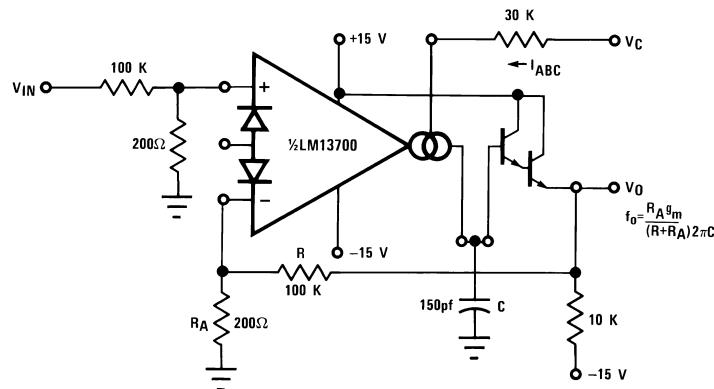
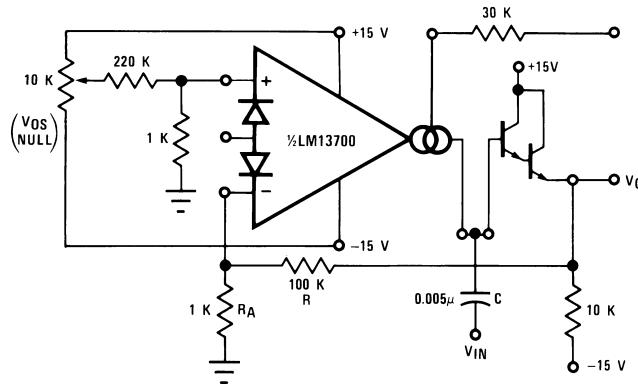


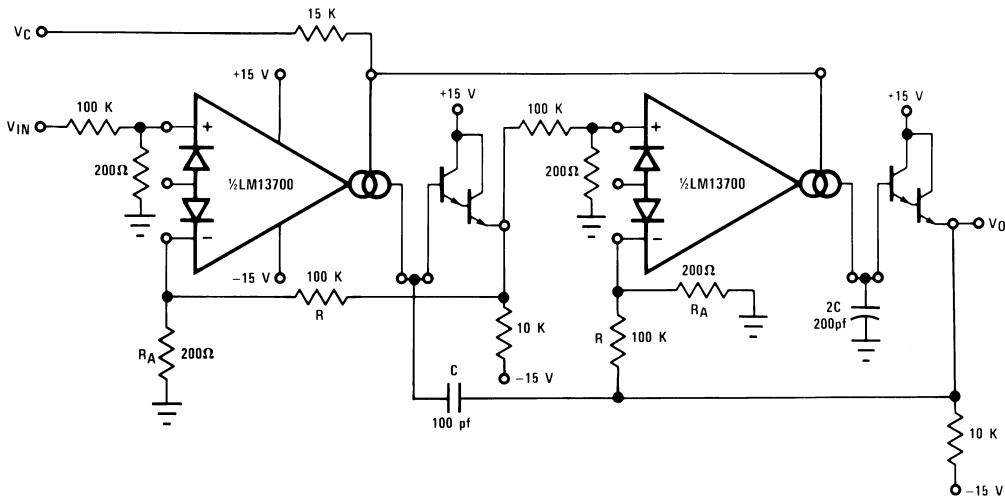
Figure 29. Voltage-Controlled Low-Pass Filter



$$f_0 = \frac{R_A g_m}{(R + R_A) 2\pi C}$$

Figure 30. Voltage-Controlled Hi-Pass Filter

System Examples (continued)



$$f_o = \frac{R_A g_m}{(R + R_A) 2\pi C}$$

Figure 31. Voltage-Controlled 2-Pole Butterworth Lo-Pass Filter

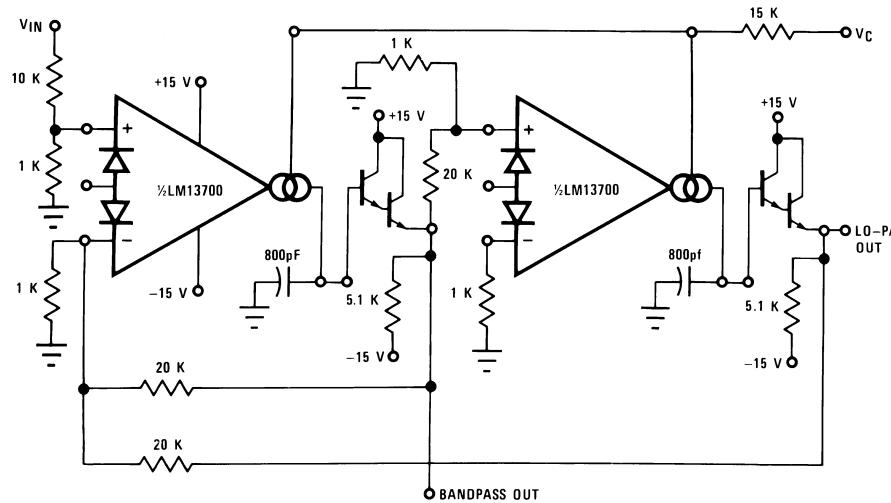


Figure 32. Voltage-Controlled State Variable Filter

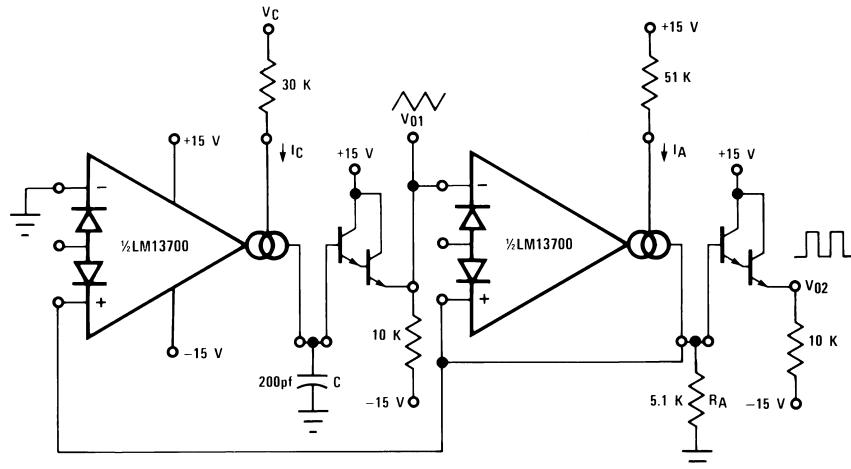
8.3.5 Voltage-Controlled Oscillators

The classic Triangular/Square Wave VCO of [Figure 33](#) is one of a variety of Voltage Controlled Oscillators which may be built utilizing the LM13700. With the component values shown, this oscillator provides signals from 200 kHz to below 2 Hz as I_C is varied from 1 mA to 10 nA. The output amplitudes are set by $I_A \times R_A$. Note that the peak differential input voltage must be less than 5 V to prevent zenering the inputs.

A few modifications to this circuit produce the ramp/pulse VCO of [Figure 34](#). When V_{O2} is high, I_F is added to I_C to increase amplifier A1's bias current and thus to increase the charging rate of capacitor C. When V_{O2} is low, I_F goes to zero and the capacitor discharge current is set by I_C .

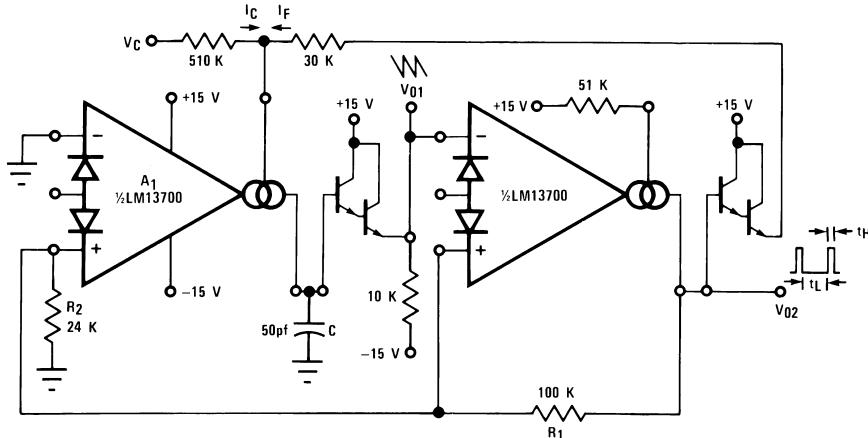
System Examples (continued)

The VC Lo-Pass Filter of Figure 29 may be used to produce a high-quality sinusoidal VCO. The circuit of Figure 34 employs two LM13700 packages, with three of the amplifiers configured as lo-pass filters and the fourth as a limiter/inverter. The circuit oscillates at the frequency at which the loop phase-shift is 360° or 180° for the inverter and 60° per filter stage. This VCO operates from 5 Hz to 50 kHz with less than 1% THD.



$$f_{OSC} = \frac{I_C}{4C_1 R_A}$$

Figure 33. Triangular/Square-Wave VCO



$$V_{PK} = \frac{(V^+ \pm 0.8V) R_2}{R_1 + R_2}$$

$$t_H \approx \frac{2V_{PK}C}{I_F}$$

$$t_L = \frac{2V_{PK}C}{I_C}$$

$$f_0 \approx \frac{I_C}{2V_{PK}C} \text{ for } I_C \ll I_F$$

Figure 34. Ramp/Pulse VCO

System Examples (continued)

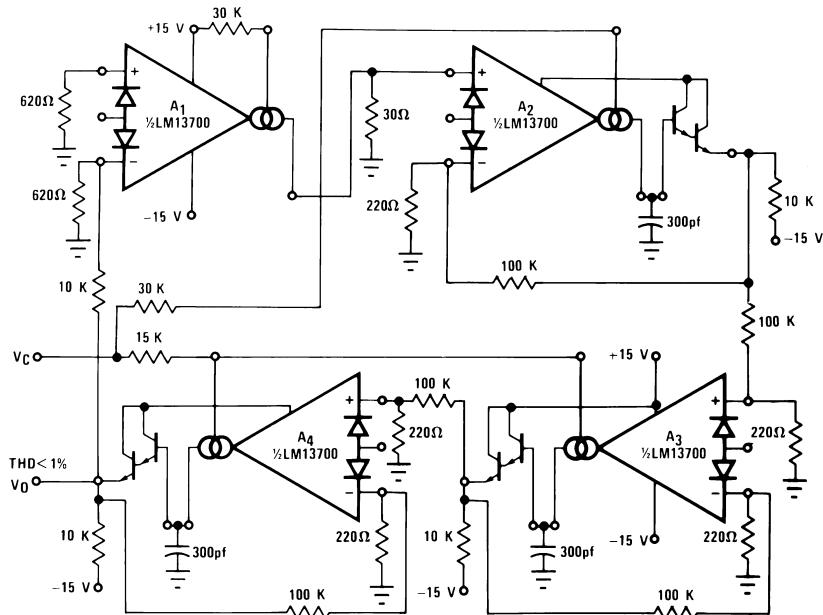


Figure 35. Sinusoidal VCO

Figure 36 shows how to build a VCO using one amplifier when the other amplifier is needed for another function.

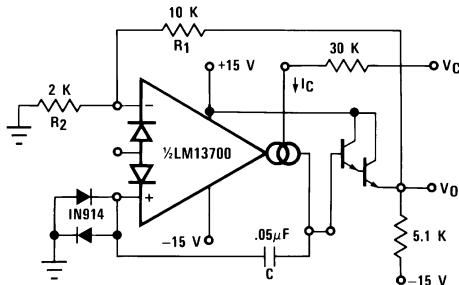


Figure 36. Single Amplifier VCO

8.3.6 Additional Applications

Figure 37 presents an interesting one-shot which draws no power supply current until it is triggered. A positive-going trigger pulse of at least 2 V amplitude turns on the amplifier through R_B and pulls the non-inverting input high. The amplifier regenerates and latches its output high until capacitor C charges to the voltage level on the non-inverting input. The output then switches low, turning off the amplifier and discharging the capacitor. The capacitor discharge rate is speeded up by shorting the diode bias pin to the inverting input so that an additional discharge current flows through D_I when the amplifier output switches low. A special feature of this timer is that the other amplifier, when biased from V_O , can perform another function and draw zero stand-by power as well.

System Examples (continued)

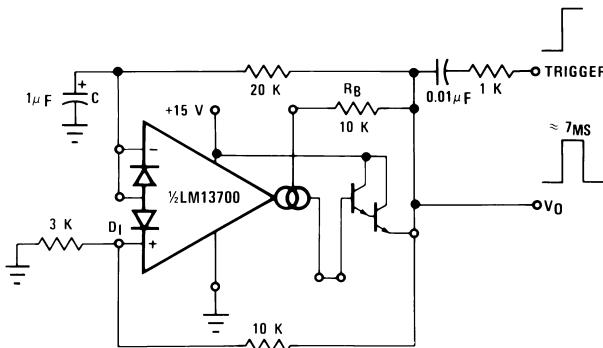


Figure 37. Zero Stand-By Power Timer

The operation of the multiplexer of Figure 38 is very straightforward. When A1 is turned on it holds V_O equal to V_{IN1} and when A2 is supplied with bias current then it controls V_O . C_C and R_C serve to stabilize the unity-gain configuration of amplifiers A1 and A2. The maximum clock rate is limited to about 200 kHz by the LM13700 slew rate into 150 pF when the $(V_{IN1}-V_{IN2})$ differential is at its maximum allowable value of 5 V.

The Phase-Locked Loop of Figure 39 uses the four-quadrant multiplier of Figure 24 and the VCO of Figure 36 to produce a PLL with a ±5% hold-in range and an input sensitivity of about 300 mV.

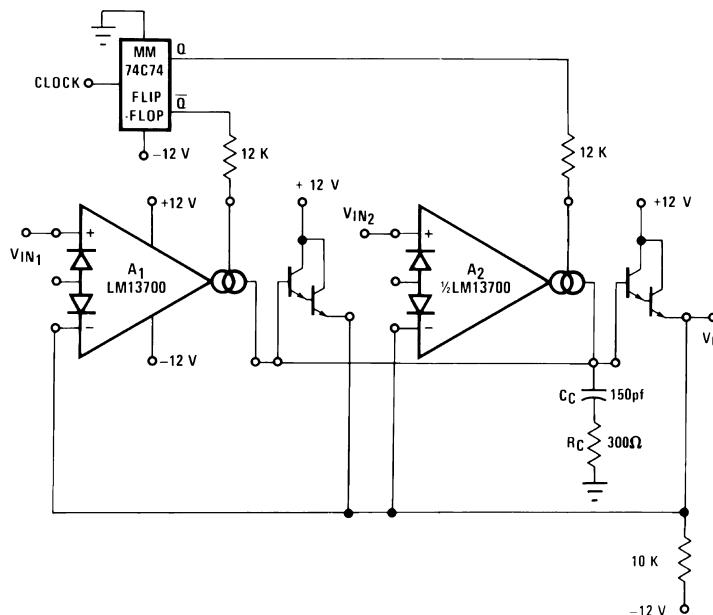


Figure 38. Multiplexer

System Examples (continued)

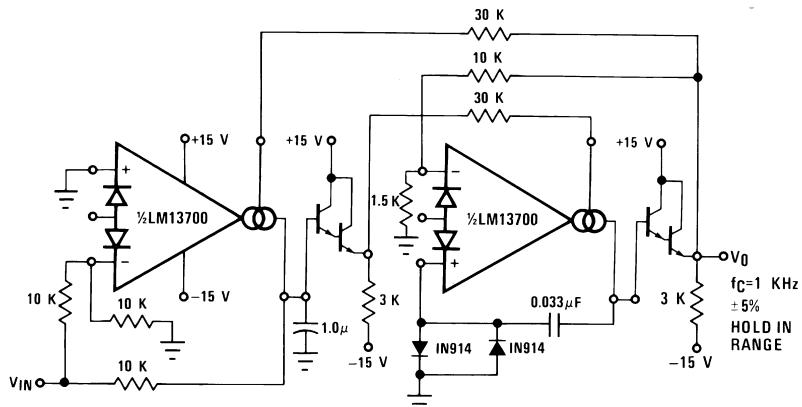


Figure 39. Phase Lock Loop

The Schmitt Trigger of [Figure 40](#) uses the amplifier output current into R to set the hysteresis of the comparator; thus $V_H = 2 \times R \times I_B$. Varying I_B will produce a Schmitt Trigger with variable hysteresis.

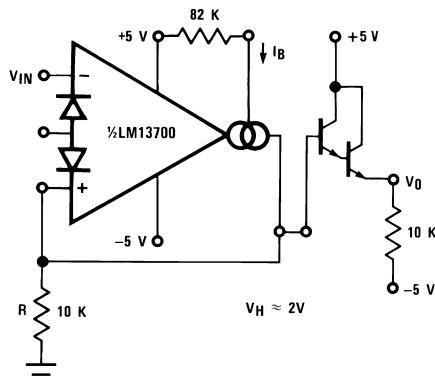


Figure 40. Schmitt Trigger

[Figure 41](#) shows a Tachometer or Frequency-to-Voltage converter. Whenever A1 is toggled by a positive-going input, an amount of charge equal to $(V_H - V_L) C_t$ is sourced into C_f and R_t . This once per cycle charge is then balanced by the current of V_o / R_t . The maximum F_{IN} is limited by the amount of time required to charge C_t from V_L to V_H with a current of I_B , where V_L and V_H represent the maximum low and maximum high output voltage swing of the LM13700. D1 is added to provide a discharge path for C_t when A1 switches low.

The Peak Detector of [Figure 42](#) uses A2 to turn on A1 whenever V_{IN} becomes more positive than V_o . A1 then charges storage capacitor C to hold V_o equal to V_{IN} PK. Pulling the output of A2 low through D1 serves to turn off A1 so that V_o remains constant.

System Examples (continued)

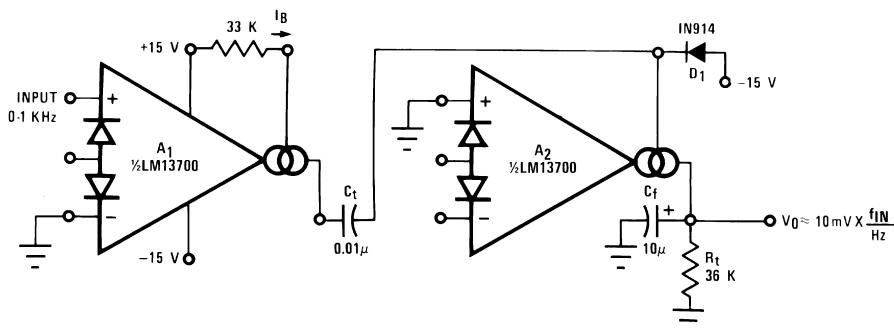


Figure 41. Tachometer

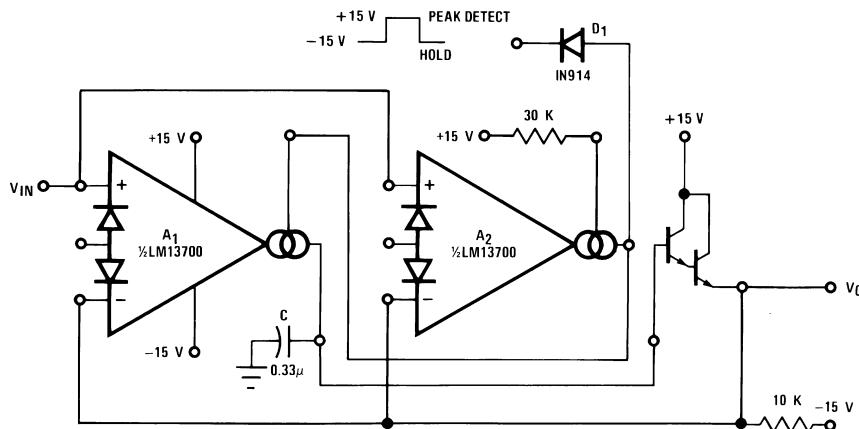


Figure 42. Peak Detector and Hold Circuit

The Ramp-and-Hold of Figure 44 sources I_B into capacitor C whenever the input to A1 is brought high, giving a ramp-rate of about 1 V/ms for the component values shown.

The true-RMS converter of Figure 45 is essentially an automatic gain control amplifier which adjusts its gain such that the AC power at the output of amplifier A1 is constant. The output power of amplifier A1 is monitored by squaring amplifier A2 and the average compared to a reference voltage with amplifier A3. The output of A3 provides bias current to the diodes of A1 to attenuate the input signal. Because the output power of A1 is held constant, the RMS value is constant and the attenuation is directly proportional to the RMS value of the input voltage. The attenuation is also proportional to the diode bias current. Amplifier A4 adjusts the ratio of currents through the diodes to be equal and therefore the voltage at the output of A4 is proportional to the RMS value of the input voltage. The calibration potentiometer is set such that V_O reads directly in RMS volts.

System Examples (continued)

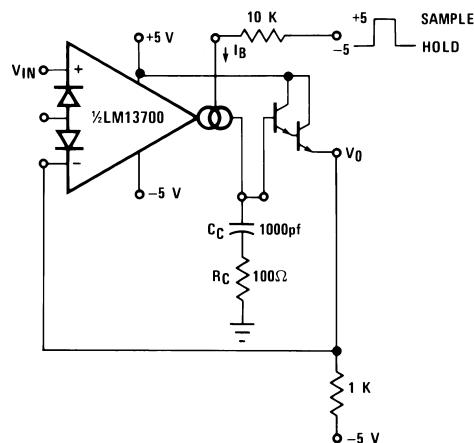


Figure 43. Sample-Hold Circuit

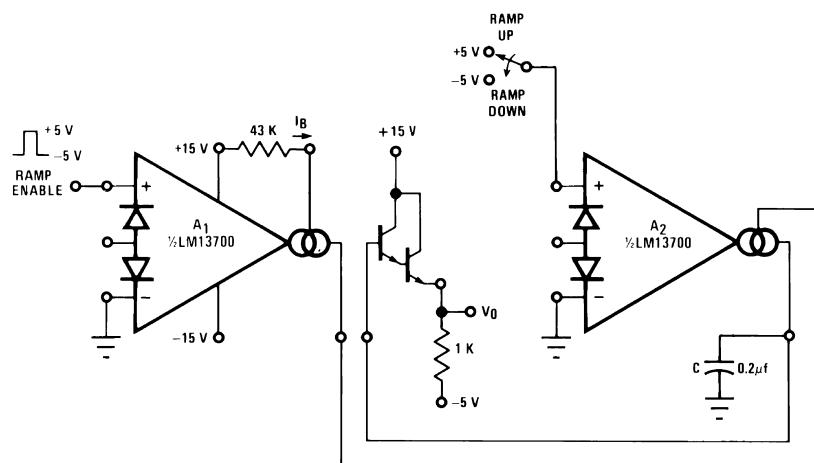


Figure 44. Ramp and Hold

System Examples (continued)

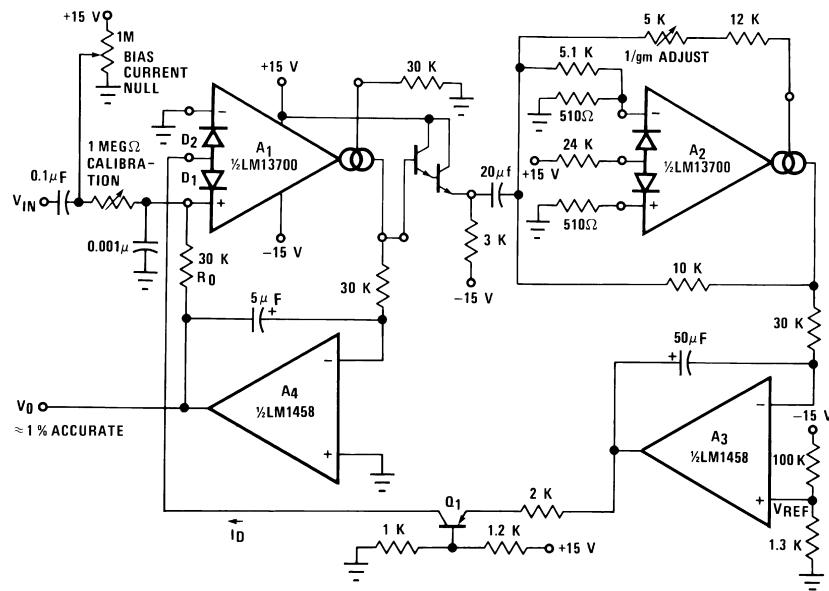


Figure 45. True RMS Converter

The circuit of Figure 46 is a voltage reference of variable Temperature Coefficient. The 100-k Ω potentiometer adjusts the output voltage which has a positive TC above 1.2 V, zero TC at about 1.2 V, and negative TC below 1.2 V. This is accomplished by balancing the TC of the A2 transfer function against the complementary TC of D1.

The wide dynamic range of the LM13700 allows easy control of the output pulse width in the Pulse Width Modulator of Figure 47.

For generating I_{ABC} over a range of 4 to 6 decades of current, the system of Figure 48 provides a logarithmic current out for a linear voltage in.

Since the closed-loop configuration ensures that the input to A2 is held equal to 0 V, the output current of A1 is equal to $I_3 = -V_C/R_C$.

The differential voltage between Q1 and Q2 is attenuated by the R1,R2 network so that A1 may be assumed to be operating within its linear range. From Equation 5, the input voltage to A1 is:

$$V_{IN1} = \frac{-2kTl_3}{qI_2} = \frac{-2kTV_C}{qI_2R_C} \quad (11)$$

The voltage on the base of Q1 is then

$$V_{B1} = \frac{(R_1 + R_2)V_{IN1}}{R_1} \quad (12)$$

The ratio of the Q1 and Q2 collector currents is defined by:

$$V_{B1} = \frac{kT}{q} \ln \frac{I_{C2}}{I_{C1}} \approx \frac{kT}{q} \ln \frac{I_{ABC}}{I_1} \quad (13)$$

Combining and solving for I_{ABC} yields:

$$I_{ABC} = I_1 \exp \frac{2(R_1 + R_2)V_C}{R_1 I_2 R_C} \quad (14)$$

This logarithmic current is used to bias the circuit of Figure 22 to provide temperature independent stereo attenuation characteristic.

System Examples (continued)

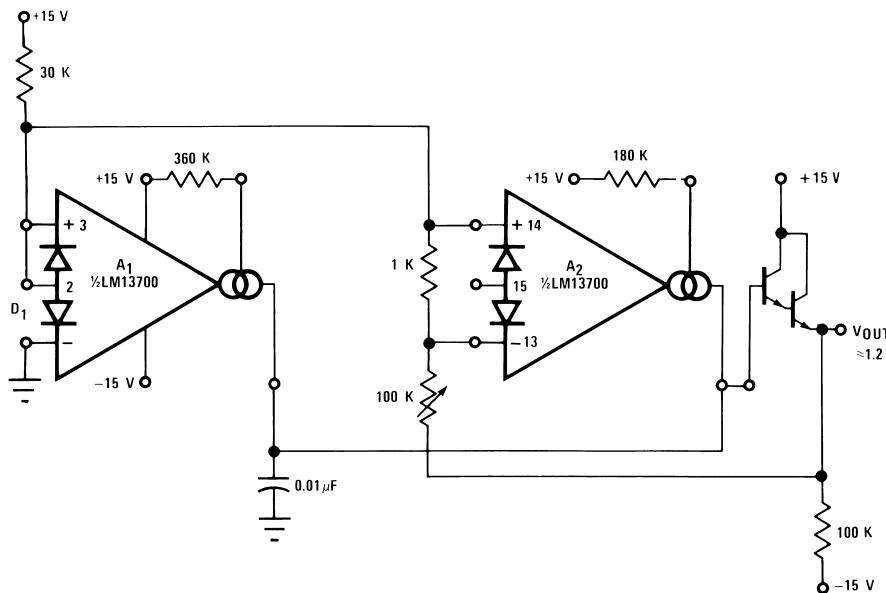


Figure 46. Delta VBE Reference

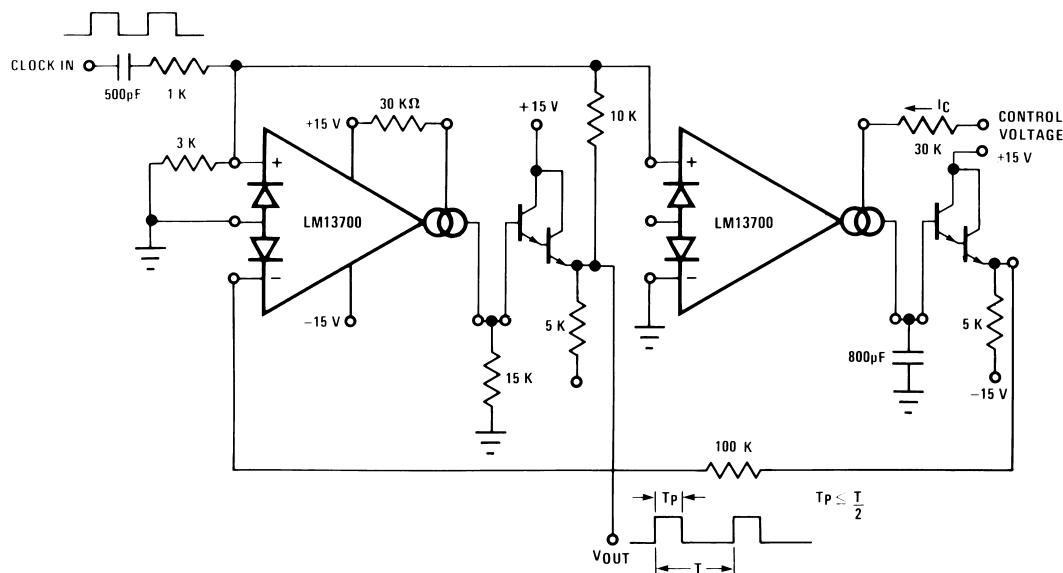
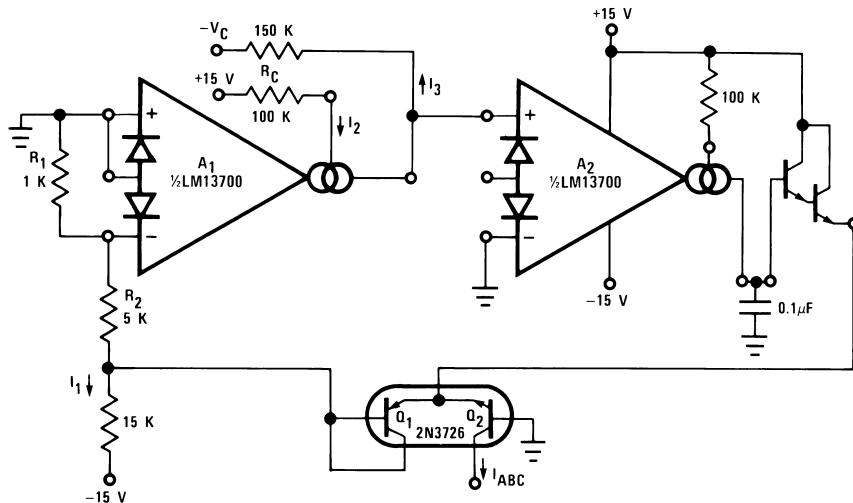


Figure 47. Pulse Width Modulator

System Examples (continued)



$$I_{ABC} = I_1 \exp \frac{-C I_3}{I_2}$$

Figure 48. Logarithmic Current Source

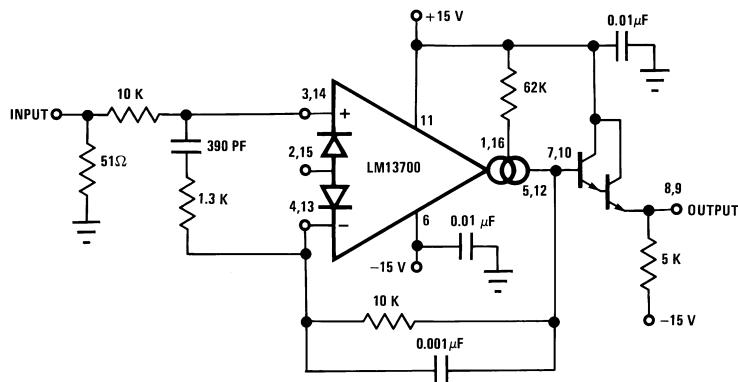


Figure 49. Unity Gain Follower

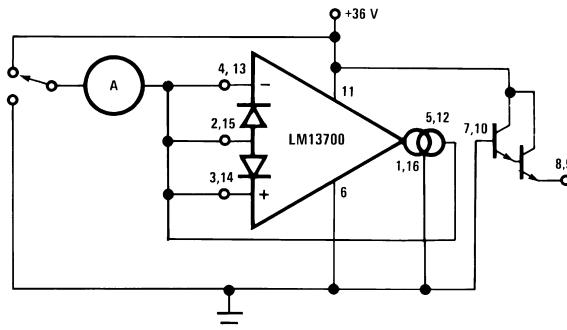
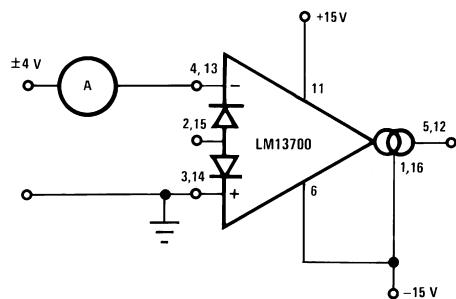


Figure 50. Leakage Current Test Circuit

System Examples (continued)**Figure 51. Differential Input Current Test Circuit**

9 Power Supply Recommendations

The LM13700 can operate with either a single-ended supply or a dual supplies. The supplies should be low impedance sources with sufficient bypassing. Use of low-ESR sufficiently rated voltage ceramic capacitors is recommended. When bypassing dual supply configurations, the supply bypass capacitors should couple to ground.

10 Layout

10.1 Layout Guidelines

Place supply bypass capacitors as close to the appropriate supply pins as possible. When multiple bypass capacitors are used, the smallest value capacitor should be closest to the supply pin.

Use of a ground plane to minimize ground impedance and provide constant signal impedance is recommended. Avoid routing signal traces over any gaps in the ground plane.

Feedback components and passives should be placed close to the device pins to minimize parasitic impedances. When using capacitors to limit bandwidth, the capacitor should be closer to the device pin than any ballasting or gain resistors.

10.2 Layout Example

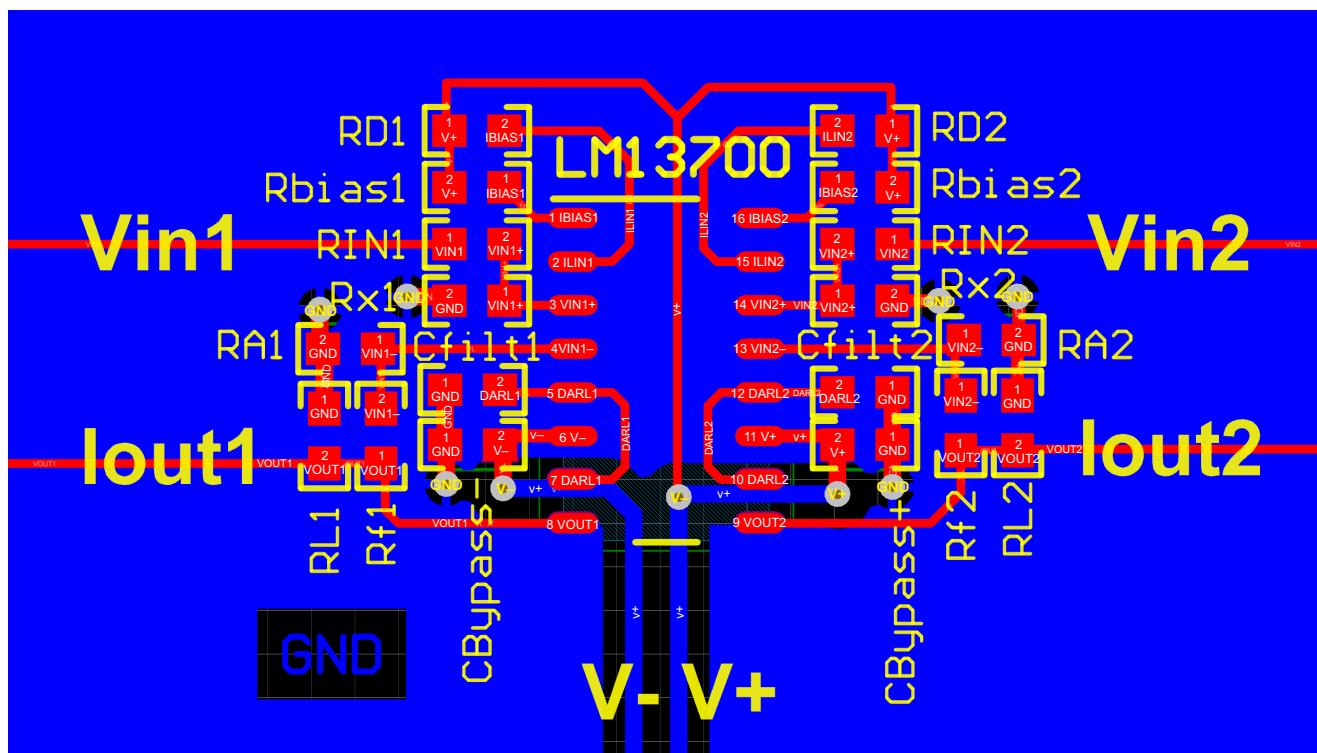


Figure 52. Layout Recommendation



Low Cost Quad Voltage Controlled Amplifier

SSM2164

FEATURES

- Four High Performance VCAs in a Single Package
- 0.02% THD
- No External Trimming
- 120 dB Gain Range
- 0.07 dB Gain Matching (Unity Gain)
- Class A or AB Operation

APPLICATIONS

- Remote, Automatic, or Computer Volume Controls
- Automotive Volume/Balance/Faders
- Audio Mixers
- Compressor/Limiters/Companders
- Noise Reduction Systems
- Automatic Gain Controls
- Voltage Controlled Filters
- Spatial Sound Processors
- Effects Processors

GENERAL DESCRIPTION

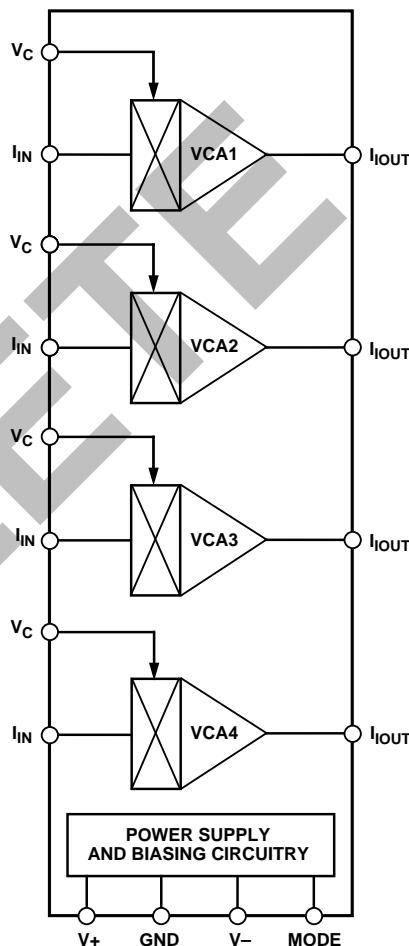
The SSM2164 contains four independent voltage controlled amplifiers (VCAs) in a single package. High performance (100 dB dynamic range, 0.02% THD) is provided at a very low cost-per-VCA, resulting in excellent value for cost sensitive gain control applications. Each VCA offers current input and output for maximum design flexibility, and a ground referenced -33 mV/dB control port.

All channels are closely matched to within 0.07 dB at unity gain, and 0.24 dB at 40 dB of attenuation. A 120 dB gain range is possible.

A single resistor tailors operation between full Class A and AB modes. The pinout allows upgrading of SSM2024 designs with minimal additional circuitry.

The SSM2164 will operate over a wide supply voltage range of ± 4 V to ± 18 V. Available in 16-pin P-DIP and SOIC packages, the device is guaranteed for operation over the extended industrial temperature range of -40°C to $+85^{\circ}\text{C}$.

FUNCTIONAL BLOCK DIAGRAM



REV. 0

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 617/329-4700 Fax: 617/326-8703

SSM2164—SPECIFICATIONS

ELECTRICAL SPECIFICATIONS ($V_s = \pm 15$ V, $A_v = 0$ dB, 0 dBu = 0.775 Vrms, $V_{IN} = 0$ dBu, $R_{IN} = R_{OUT} = 30$ k Ω , $f = 1$ kHz, $-40^\circ\text{C} < T_A < +85^\circ\text{C}$ using Typical Application Circuit (Class AB), unless otherwise noted. Typical specifications apply at $T_A = +25^\circ\text{C}$.)

Parameter	Conditions	SSM2164			Units
		Min	Typ	Max	
AUDIO SIGNAL PATH					
Noise	$V_{IN} = \text{GND}$, 20 kHz Bandwidth	-94			dBu
Headroom	Clip Point = 1% THD+N	22			dBu
Total Harmonic Distortion	2nd and 3rd Harmonics Only				%
	$A_v = 0$ dB, Class A	0.02	.1		%
	$A_v = \pm 20$ dB, Class A ¹	0.15			%
	$A_v = 0$ dB, Class AB	0.16			%
	$A_v = \pm 20$ dB, Class AB ¹	0.3			%
Channel Separation	$C_F = 10$ pF	-110			dB
Unity Gain Bandwidth	$C_F = 10$ pF	500			kHz
Slew Rate		0.7			mA/ μ s
Input Bias Current		± 10			nA
Output Offset Current		± 50			nA
Output Compliance	$V_{IN} = 0$	± 0.1			V
CONTROL PORT					
Input Impedance	(Note 2)	5			k Ω
Gain Constant		-33			mV/dB
Gain Constant Temperature Coefficient		-3300			ppm/ $^\circ\text{C}$
Control Feedthrough	0 dB to -40 dB Gain Range ³	1.5	8.5		mV
Gain Matching, Channel-to-Channel	$A_v = 0$ dB	0.07			dB
	$A_v = -40$ dB	0.24			dB
Maximum Attenuation		-100			dB
Maximum Gain		+20			dB
POWER SUPPLIES					
Supply Voltage Range	Class AB	± 4			V
Supply Current		6	18		mA
Power Supply Rejection Ratio	60 Hz	90	8		dB

NOTES

¹-10 dBu input @ 20 dB gain; +10 dBu input @ -20 dB gain.

²After 60 seconds operation.

³+25°C to +85°C.

Specifications subject to change without notice.

TYPICAL APPLICATION AND TEST CIRCUIT

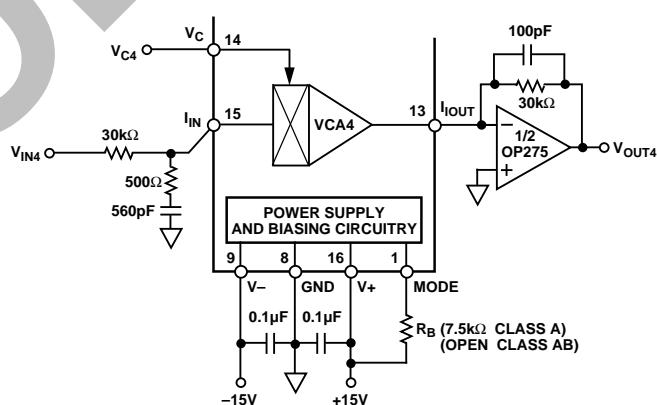


Figure 1. $R_{IN} = R_{OUT} = 30$ k Ω , $C_F = 100$ pF. Optional $R_B = 7.5$ k Ω , Biases Gain Core to Class A Operation. For Class AB, Omit R_B .

ABSOLUTE MAXIMUM RATINGS

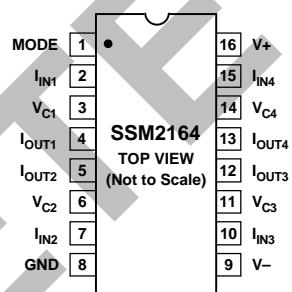
Supply Voltage ±18 V
 Input, Output, Control Voltages V₋ to V₊
 Output Short Circuit Duration to GND Indefinite
 Storage Temperature Range -65°C to +150°C
 Operating Temperature Range -40°C to +85°C
 Junction Temperature Range -65°C to +150°C
 Lead Temperature Range (Soldering 60 sec) +300°C

Package Type	θ _{JA} *	θ _{JC}	Units
16-Pin Plastic DIP (P Suffix)	76	33	°C/W
16-Pin SOIC (S Suffix)	92	27	°C/W

*θ_{JA} is specified for the worst case conditions; i.e., θ_{JA} is specified for device in socket for P-DIP packages, θ_{JA} is specified for device soldered in circuit board for SOIC package.

ORDERING GUIDE

Model	Temperature Range	Package Description	Package Options
SSM2164P	-40°C to +85°C	Plastic DIP	N-16
SSM2164S	-40°C to +85°C	Narrow SOIC	R-16A

PIN CONFIGURATION**16-Lead Epoxy DIP and SOIC****CAUTION**

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the SSM2164 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



SSM2164

Typical Performance Characteristics

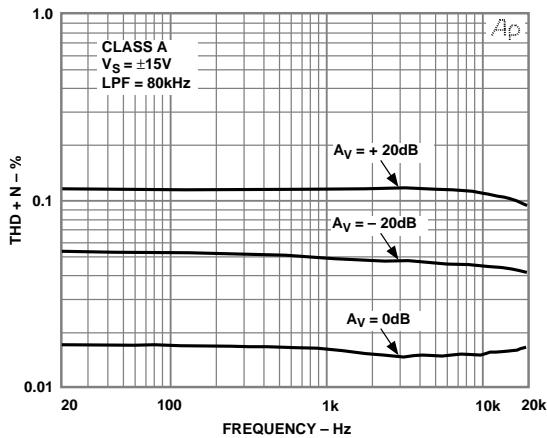


Figure 2. THD+N vs. Frequency, Class A

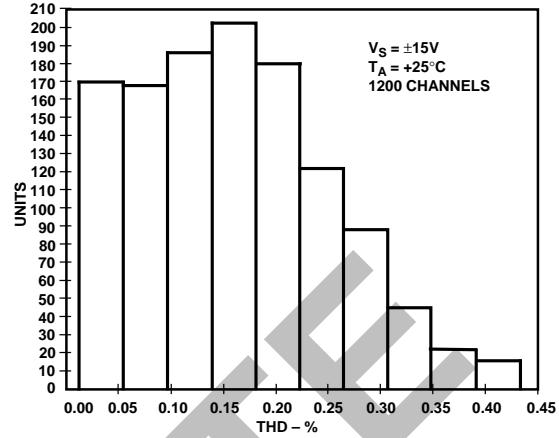


Figure 5. THD Distribution, Class AB

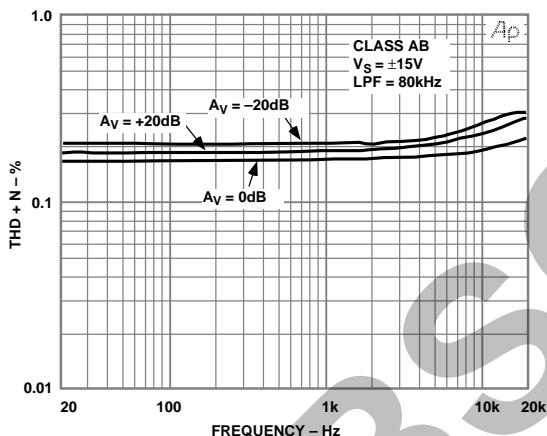


Figure 3. THD+N vs. Frequency Class, AB

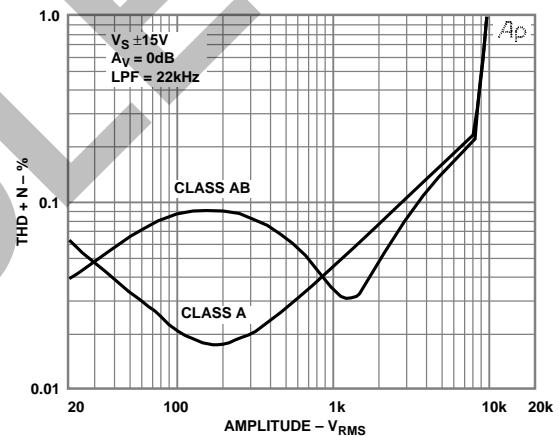


Figure 6. THD+N vs. Amplitude

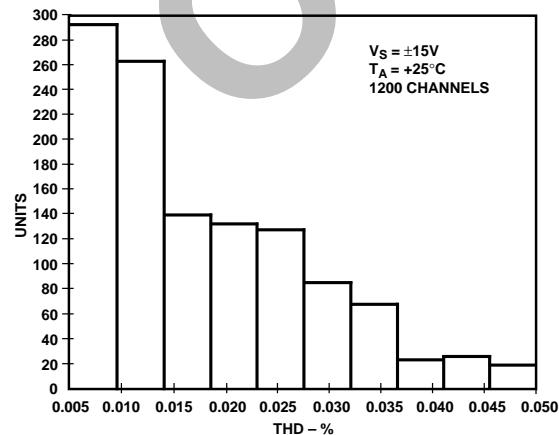


Figure 4. THD Distribution, Class A

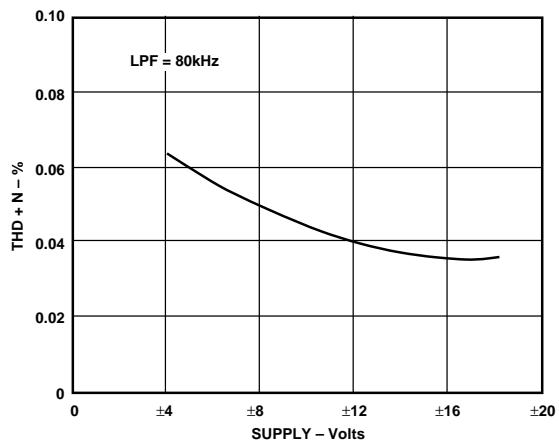
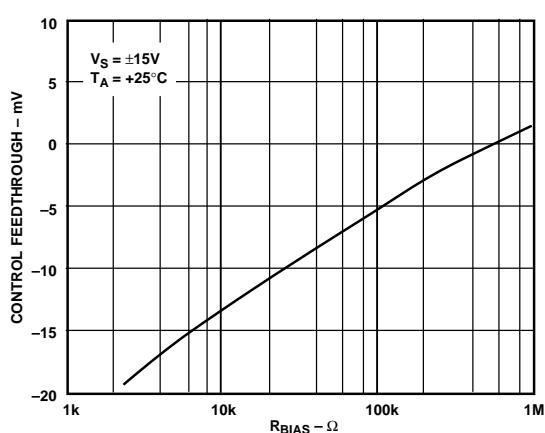
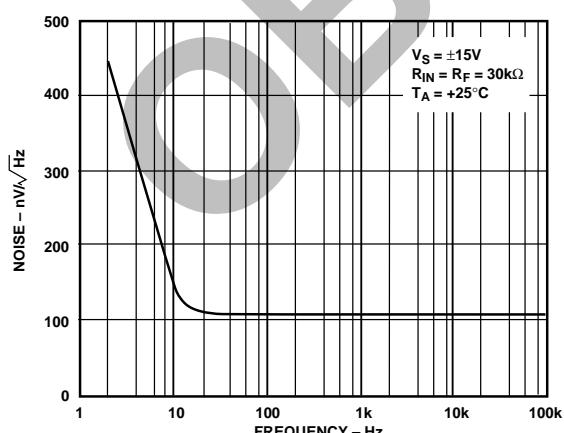
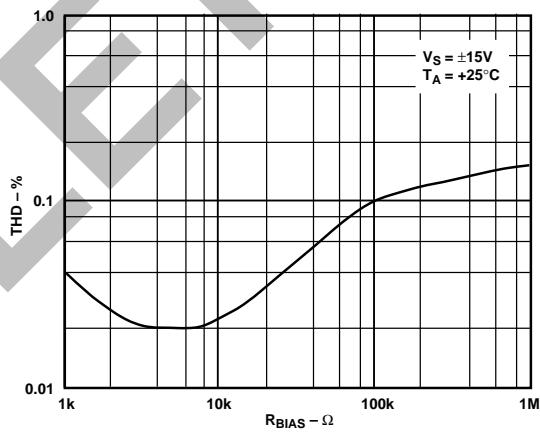
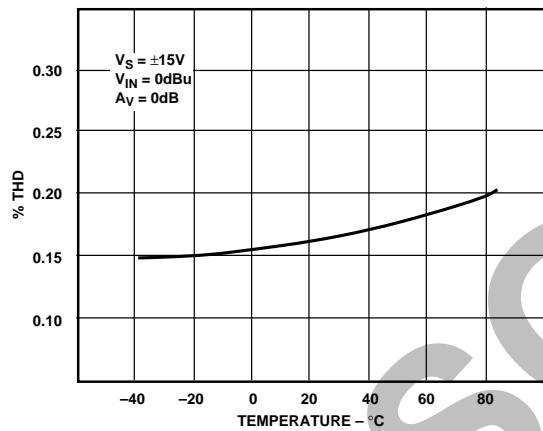
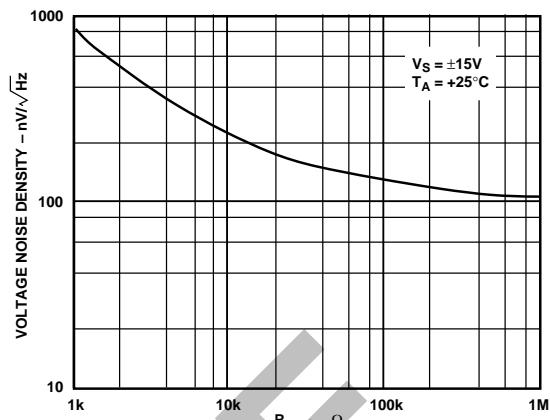
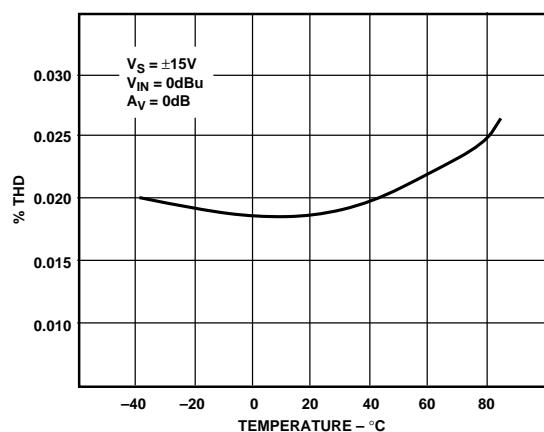


Figure 7. THD+N vs. Supply Voltage, Class A



SSM2164

Typical Performance Characteristics

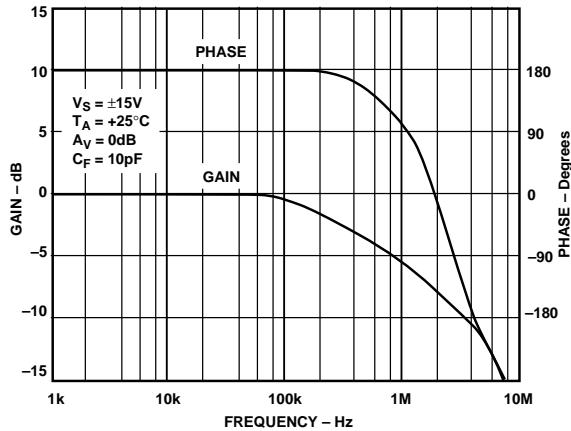


Figure 14. Gain/Phase vs. Frequency

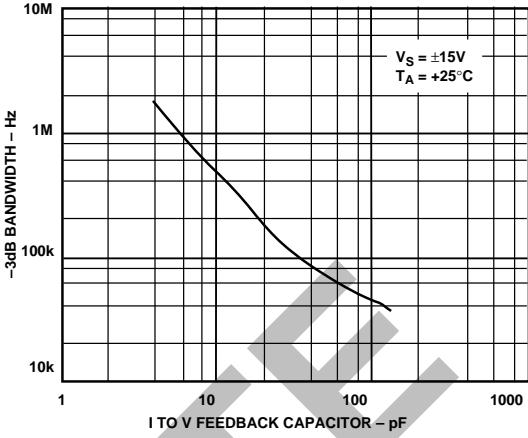


Figure 17. -3 dB Bandwidth vs. I-to-V Feedback Capacitor

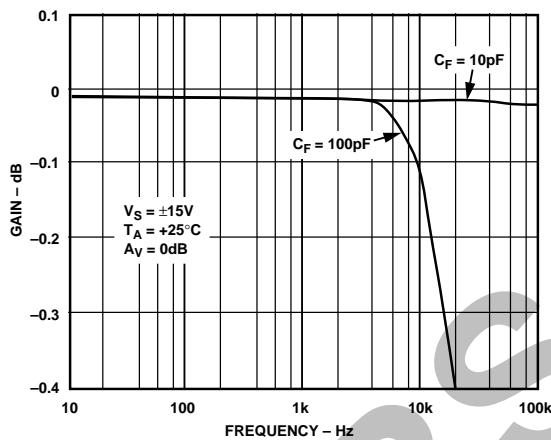


Figure 15. Gain Flatness vs. Frequency

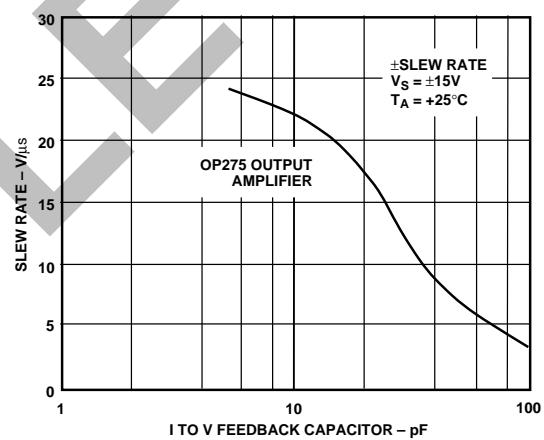


Figure 18. Slew Rate vs. I-to-V Feedback Capacitor

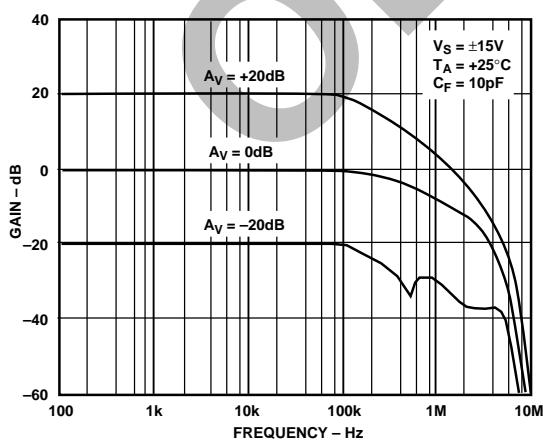


Figure 16. Bandwidth vs. Gain

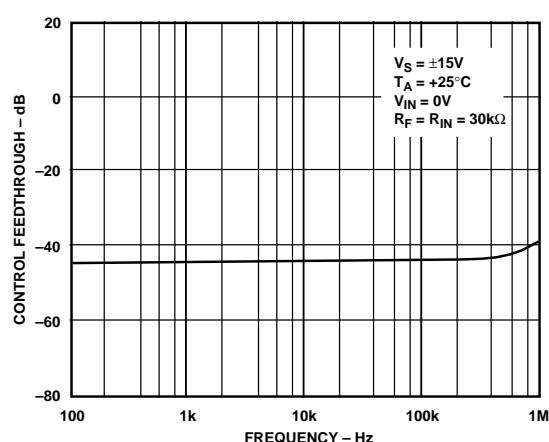


Figure 19. Control Feedthrough vs. Frequency

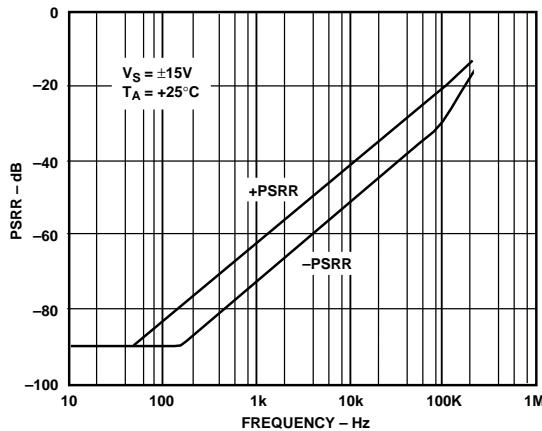


Figure 20. PSRR vs. Frequency

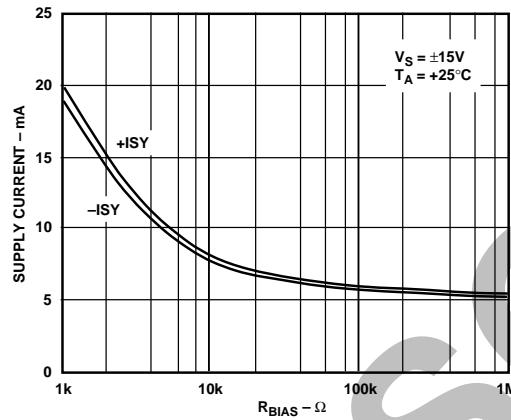
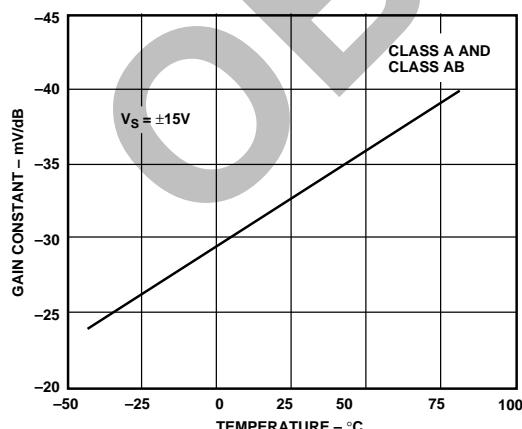
Figure 21. Supply Current vs. R_{BIAS} 

Figure 22. Gain Constant vs. Temperature

APPLICATIONS INFORMATION

Circuit Description

The SSM2164 is a quad Voltage Controlled Amplifier (VCA) with 120 dB of gain control range. Each VCA is a current-in, current-out device with a separate $-33 \text{ mV}/\text{dB}$ voltage input control port. The class of operation (either Class A or Class AB) is set by a single external resistor allowing optimization of the distortion versus noise tradeoff for a particular application. The four independent VCAs in a single 16-pin package make the SSM2164 ideal for applications where multiple volume control elements are needed.

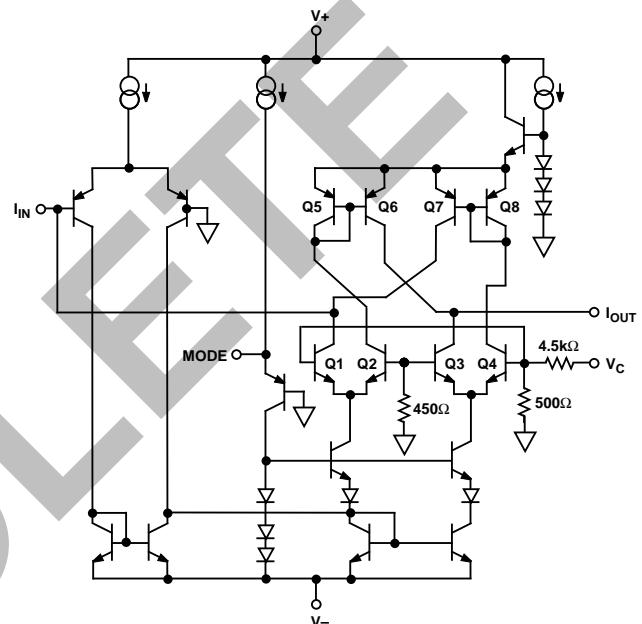


Figure 23. Simplified Schematic (One Channel)

The simplified schematic in Figure 23 shows the basic structure of one of the four VCAs in the device. The gain core is comprised of the matched differential pairs Q1-Q4 and the current mirrors of Q5, Q6 and Q7, Q8. The current input pin, I_{IN} , is connected to the collectors of Q1 and Q7, and the difference in current between these two transistors is equivalent to I_{IN} . For example, if 100 μA is flowing into the input, Q1's collector current will be 100 μA higher than Q7's collector current.

Varying the control voltage V_C , steers the signal current from one side of each differential pair to the other, resulting in either gain or attenuation. For example, a positive voltage on V_C steers more current through Q1 and Q4 and decreases the current in Q2 and Q3. The current output pin, I_{OUT} , is connected to the collector of Q3 and the current mirror (Q6) from Q2. With less current flowing through these two transistors, less current is available at the output. Thus, a positive V_C attenuates the input and a negative V_C amplifies the input. The VCA has unity gain for a control voltage of 0.0 V where the signal current is divided equally between the gain core differential pairs.

The MODE pin allows the setting of the quiescent current in the gain core of the VCA to trade off the SSM2164's THD and noise performance to an optimal level for a particular application. Higher current through the core results in lower distortion

SSM2164

but higher noise, and the opposite is true for less current. The increased noise is due to higher current noise in the gain core transistors as their operating current is increased. THD has the opposite relationship to collector current. The lower distortion is due to the decrease in the gain core transistors' emitter impedance as their operating current increases.

This classical tradeoff between THD and noise in VCAs is usually expressed as the choice of using a VCA in either Class A or Class AB mode. Class AB operation refers to running a VCA with less current in the gain core, resulting in lower noise but higher distortion. More current in the core corresponds to Class A performance with its lower THD but higher noise. Figures 11 and 12 show the THD and noise performance of the SSM2164 as the bias current is adjusted. Notice the two characteristics have an inverse characteristic.

The quiescent current in the core is set by adding a single resistor from the positive supply to the MODE pin. As the simplified schematic shows, the potential at the MODE pin is one diode drop above the ground pin. Thus, the formula for the MODE current is:

$$I_{MODE} = \frac{(V+) - 0.6 V}{R_B}$$

With ± 15 V supplies, an R_B of $7.5\text{k}\Omega$ gives Class A biasing with a current of 1.9 mA . Leaving the MODE pin open sets the SSM2164 in Class AB with $30\text{ }\mu\text{A}$ of current in the gain core.

Basic VCA Configuration

Figure 24 shows the basic application circuit for the SSM2164. Each of the four channels is configured identically. A $30\text{ k}\Omega$ resistor converts the input voltage to an input current for the VCA. Additionally, a $500\text{ }\Omega$ resistor in series with a 560 pF capacitor must be added from each input to ground to ensure stable operation. The output current pin should be maintained at a virtual ground using an external amplifier. In this case the OP482 quad JFET input amplifier is used. Its high slew rate, wide bandwidth, and low power make it an excellent choice for the current-to-voltage converter stage. A $30\text{ k}\Omega$ feedback resistor is chosen to match the input resistor, giving unity gain for a 0.0 V control voltage. The 100 pF capacitors ensure stability and reduce high frequency noise. They can be increased to reduce the low pass cutoff frequency for further noise reduction.

For this example, the control voltage is developed using a $100\text{ k}\Omega$ potentiometer connected between $+5\text{ V}$ and ground. This configuration results in attenuation only. To produce both gain and attenuation, the potentiometer should be connected between a positive and negative voltage. The control input has an impedance of $5\text{ k}\Omega$. Because of this, any resistance in series with V_C will attenuate the control signal. If precise control of the gain and attenuation is required, a buffered control voltage should be used.

Notice that a capacitor is connected from the control input to ground. Because the control port is connected directly to the gain core transistors, any noise on the V_C pin will increase the output noise of the VCA. Filtering the control voltage ensures that a minimal amount of noise is introduced into the VCA, allowing its full performance to be realized. In general, the largest possible capacitor value should be used to set the filter at

a low cutoff frequency. The main exception to this is in dynamic processing applications, where faster attack or decay times may be needed.

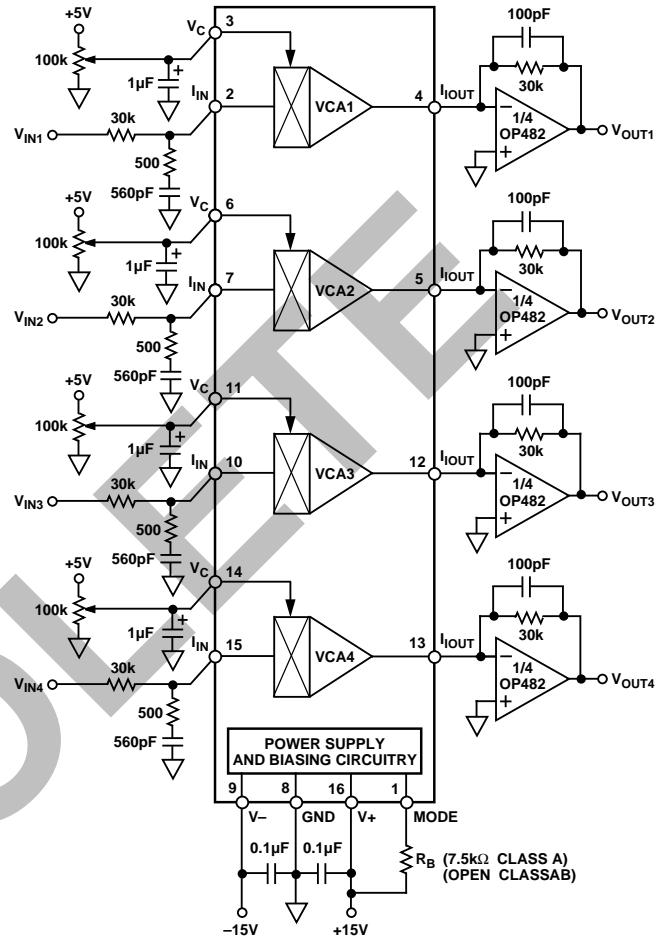


Figure 24. Basic Quad VCA Configuration

Low Cost, Four-Channel Mixer

The four VCAs in a single package can be configured to create a simple four-channel mixer as shown in Figure 25. The inputs and control ports are configured the same as for the basic VCA, but the outputs are summed into a single output amplifier. The OP176 is an excellent amplifier for audio applications because of its low noise and distortion and high output current drive. The amount of signal from each input to the common output can be independently controlled using up to 20 dB of gain or as much as 100 dB of attenuation. Additional SSM2164s could be added to increase the number of mixer channels by simply summing their outputs into the same output amplifier. Another possible configuration is to use a dual amplifier such as the OP275 to create a stereo, two channel mixer with a single SSM2164.

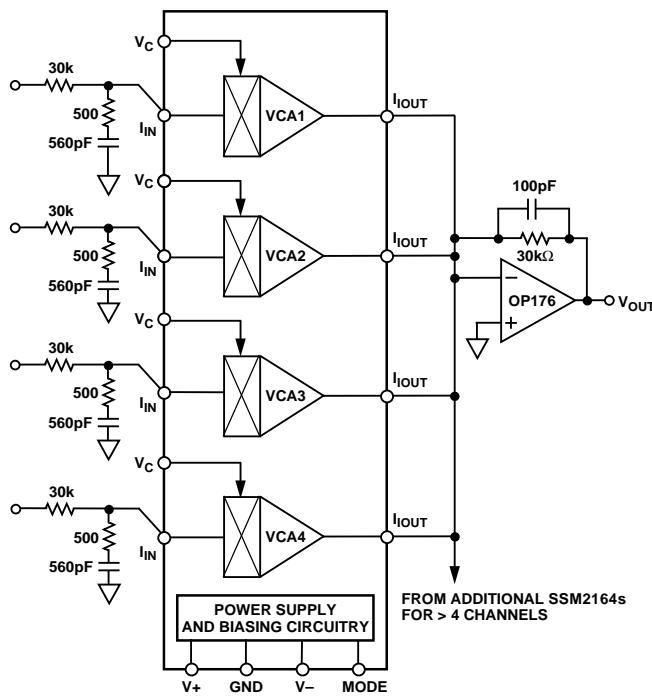


Figure 25. Four-Channel Mixer (4 to 1)

If additional SSM2164s are added, the 100 pF capacitor may need to be increased to ensure stability of the output amplifier. Most op amps are sensitive to capacitance on their inverting inputs. The capacitance forms a pole with the feedback resistor, which reduces the high frequency phase margin. As more SSM2164's are added to the mixer circuit, their output capacitance and the parasitic trace capacitance add, increasing the overall input capacitance. Increasing the feedback capacitor will maintain the stability of the output amplifier.

Digital Control of the SSM2164

One option for controlling the gain and attenuation of the SSM2164 is to use a voltage output digital-to-analog converter such as the DAC8426 (Figure 26), whose 0 V to +10 V output controls the SSM2164's attenuation from 0 dB to -100 dB. Its simple 8-bit parallel interface can easily be connected to a microcontroller or microprocessor in any digitally controlled system. The voltage output configuration of the DAC8426 provides a low impedance drive to the SSM2164 so the attenuation can be controlled accurately. The 8-bit resolution of the DAC and its full-scale voltage of +10 V gives an output of 3.9 mV/bit. Since the SSM2164 has a -33 mV/dB gain constant, the overall control law is 0.12 dB/bit or approximately 8 bits/dB. The input and output configuration for the SSM2164 is the same as for the basic VCA circuit shown earlier. The 4-to-1 mixer configuration could also be used.

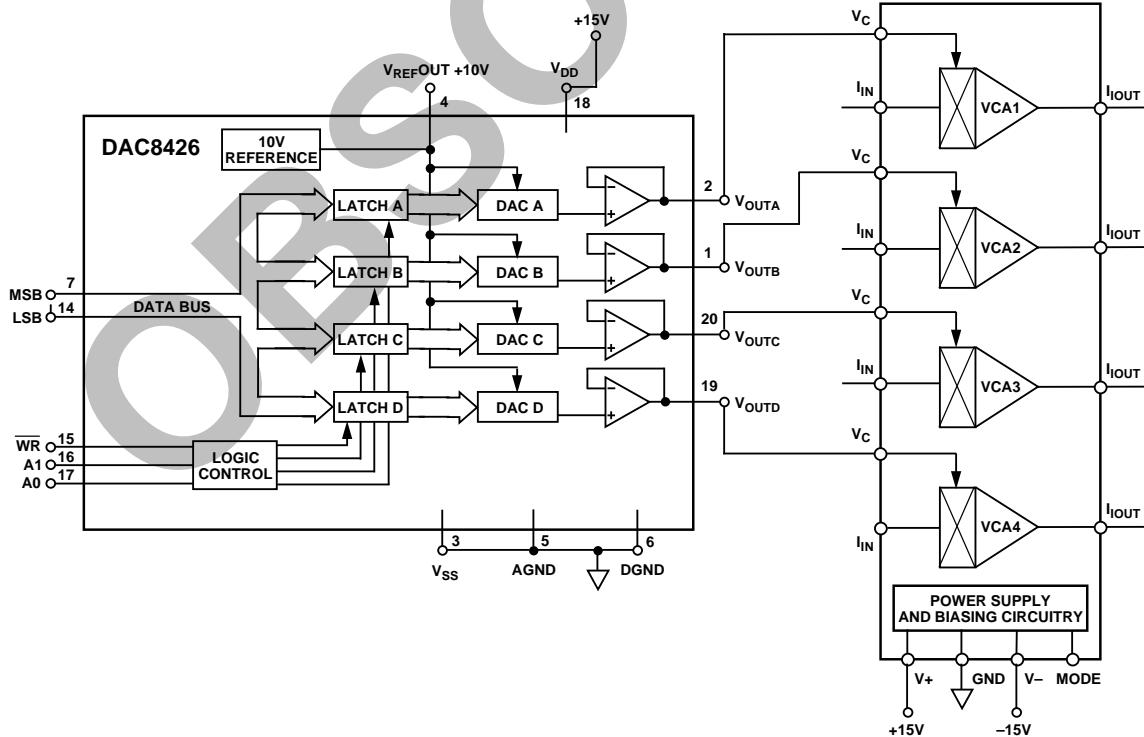


Figure 26. Digital Control of VCA Gain

SSM2164

Single Supply Operation

The SSM2164 can easily be operated from a single power supply as low as +8 V or as high as +36 V. The key to using a single supply is to reference all ground connections to a voltage midway between the supply and ground as shown in Figure 27. The OP176 is used to create a pseudo-ground reference for the SSM2164. Both the OP482 and OP176 are single supply amplifiers and can easily operate over the same voltage range as the SSM2164 with little or no change in performance.

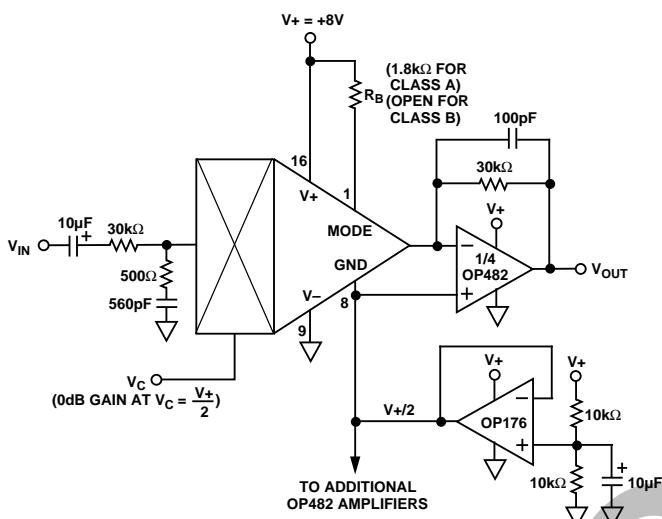


Figure 27. Single Supply Operation of the SSM2164
(One Channel Shown)

The reference voltage is set by the resistor divider from the positive supply. Two 10 k Ω resistors create a voltage equal to the positive supply divided by 2. The 10 μ F capacitor filters the supply voltage, providing a low noise reference to the circuit. This reference voltage is then connected to the GND pin of the SSM2164 and the noninverting inputs of all the output amplifiers. It is important to buffer the resistor divider with the OP176 to ensure a low impedance pseudo-ground connection for the SSM2164.

The input can either be referenced to this same mid-supply voltage or ac coupled as is done in this case. If the entire system is single supply, then the input voltage will most likely already be referenced to the midpoint; if this is the case, the 10 μ F input capacitor can be eliminated. Unity gain is set when V_C equals the voltage on the GND pin. Thus, the control voltage should also be referenced to the same midsupply voltage.

The value of the MODE setting resistor may also change depending on the total supply voltage. Because the GND pin is at a pseudo-ground potential, the equation to set the MODE current now becomes:

$$I_{MODE} = \frac{(V+) - V_{GND} - 0.6\text{ V}}{R_B}$$

The value of 1.8 k Ω results in Class A biasing for the case of using a +8 V supply.

Upgrading SSM2024 Sockets

The SSM2164 is intended to replace the SSM2024, an earlier generation quad VCA. The improvements in the SSM2164 have resulted in a part that is not a drop-in replacement to the SSM2024, but upgrading applications with the SSM2024 is a simple task. The changes are shown in Figure 28. Both parts have identical pinouts with one small exception. The MODE input (Pin 1) does not exist on the SSM2024. It has fixed internal biasing, whereas flexibility was designed into the SSM2164. A MODE set resistor should be added for Class A operation, but if the SSM2164 is going to be operated in Class AB, no external resistor is needed.

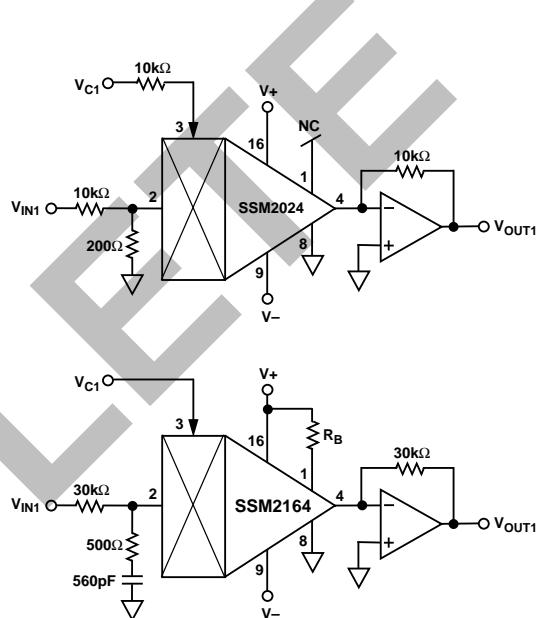


Figure 28. Upgrading SSM2024 Sockets with SSM2164

Since both parts are current output devices, the output configuration is nearly identical, except that the 10 k Ω resistors should be increased to 30 k Ω to operate the SSM2164 in its optimum range. The 10 k Ω input resistor for the SSM2024 should also be increased to 30 k Ω to match the output resistor. Additionally, the 200 Ω resistor should be replaced by a 500 Ω resistor in series with 560 pF for the SSM2164 circuit.

One last change is the control port configuration. The SSM2024's control input is actually a current input. Thus, a resistor was needed to change the control voltage to a current. This resistor should be removed for the SSM2164 to provide a direct voltage input. In addition, the SSM2024 has a log/log control relationship in contrast to the SSM2164's linear/log gain constant. The linear input is actually much easier to control, but the difference may necessitate adjusting a SSM2024 based circuit's control voltage gain curve. By making these relatively simple changes, the superior performance of the SSM2164 can easily be realized.



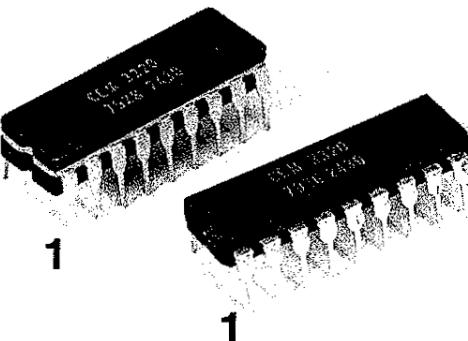
CURTIS ELECTROMUSIC SPECIALTIES

CEM 3320

Voltage Controlled Filter

The CEM 3320 is a high performance voltage controlled four-pole filter complete with on-chip voltage controllable resonance. The four independent sections may be interconnected to provide a wide variety of filter responses, such as low pass, high pass, band pass and all pass. A single input exponentially controls the frequency over greater than a ten octave range with little control voltage feed-through. Another input controls the resonance in a modified linear manner from zero to low distortion oscillation. For those

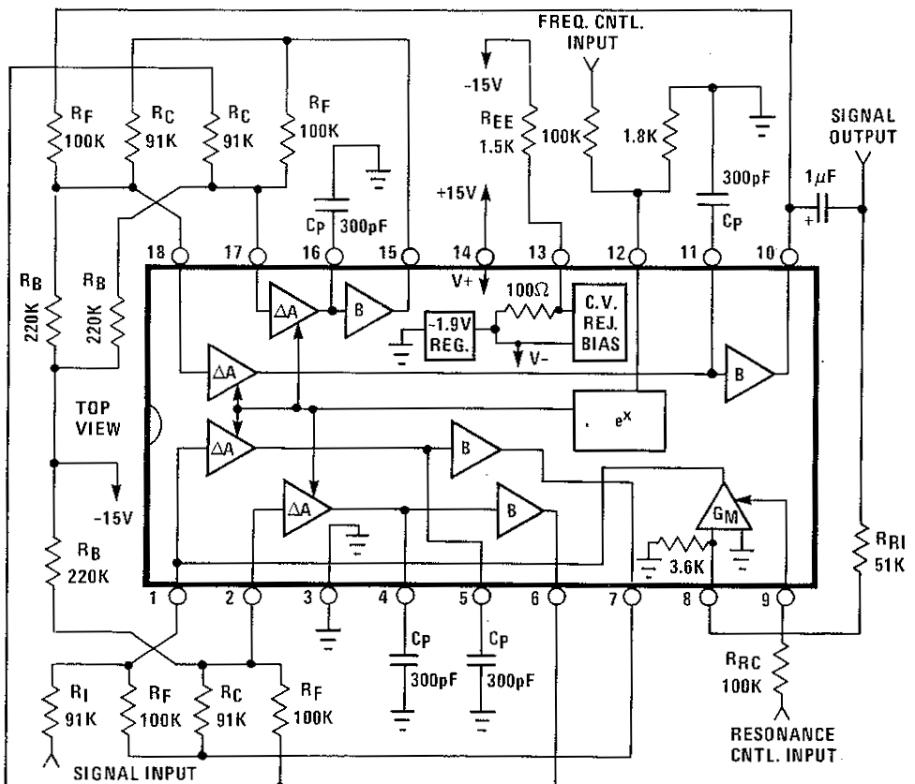
demanding applications, provision has been made to allow trimming for improved control voltage rejection. Each filter section features a novel variable gain cell which, unlike the traditional cell, is fully temperature compensated, exhibits a better signal-to-noise ratio and generates its low distortion predominantly in the second harmonic. The device further includes a minus two volt regulator to ensure low power dissipation and consequent low warm-up drift even with ± 15 volt supplies.



1

1

Circuit Block and Connection Diagram



Features

- Low Cost
- Voltage Controllable Frequency: 12 octave range minimum
- Voltage Controllable Resonance: From zero to oscillation
- Accurate Exponential Frequency Scale
- Accurate Linear Resonance Scale
- Low Control Voltage Feed-through: -45dB typical
- Filter Configurable into Low Pass, High Pass, All Pass, etc.
- Large Output: 12V.P.P. typical
- Low Noise: -86dB typical
- Low Distortion in Passband: 0.1% typical
- Low Warm Up Drift
- Configurable into Low Distortion Voltage Controlled Sine Wave Oscillator
- ± 15 Volt Supplies

CEM 3320

Electrical Characteristics

Application Hints

$V_{CC} = +15V$		$R_F = 100K$		$T_A = 25^\circ C$	
Parameter		Min.	Typ.	Max.	Units
Pole Frequency Control Range		3500:1	10,000:1	—	
Sensitivity of Pole Frequency Control Scale, Midrange		57.5	60	62.5	mV/decade
Tempco of Pole Frequency Control Scale		3000	3300	3600	ppm
Exponential Error of Pole Frequency Control Scale ¹		—	4	12	%
Gain of Variable Gain Cell at $V_C=0$		0.7	0.9	1.3	
Max Gain of Variable Gain Cell		2.4	3.0	3.6	
Tempco of Variable Gain Cell ²		—	500	1500	ppm
Output Impedance of Gain Cell ²		0.5	1.0	2.0	MΩ
Pole Frequency Control Feedthrough		—	60	200	mV
Pole Frequency Warm-up Drift		—	.5	1.5	%
Gm of Resonance Control Element at $I_{CR}=100\mu A$.8	1.0	1.2	mmhos
Amount of Resonance Obtainable Before Oscillation		20	30	—	dB
Resonance Control Feedthrough ³		—	0.2	1.5	V
Output Swing At Clipping		10	12	14	V.P.P.
Output Noise re Max Output ⁴		-76	-86	—	dB
Rejection in Bandreject		73	83	—	dB
Distortion in Passband ^{5,7}		—	0.1	0.3	%
Distortion in Bandreject ^{6,7}		—	0.3	1	%
Distortion of Sine Wave Oscillation ⁸		—	0.5	1.5	%
Internal Reference Current, I_{REF}		45	63	85	μA
Input Bias Current of Frequency Control Input		0.2	0.5	1.5	μA
Input Impedance to Resonance Signal Input		2.7	3.6	4.5	KΩ
Buffer Slew Rate		1.5	3.0	—	V/uS
Buffer Input Bias Current ($ I_{EE} =8mA$)		±8	±30	±100	nA
Buffer Sink Capability		.4	.5	.63	mA
Buffer Output Impedance ²		75	100	200	Ω
Positive Supply Range		+9	—	+18	V
Negative Supply Range ⁹		-4	—	-18	V
Positive Supply Current		3.8	5	6.5	mA

Note 1: $-25mV < V_C < +155mV$. Most of this error occurs in upper two octaves.

Note 2: $V_C = 0$

Note 3: Untrimmed, $0 < I_{CR} < 100\mu A$

Note 4: Filter is connected as low pass and set for 20 KHz cut-off frequency.

Note 5: Output signal is 3dB below clipping point.

Note 6: Output signal is 3dB below passband level, which is 3dB below clipping point. In general, this is worst case condition.

Note 7: Distortion is predominantly second harmonic.

Note 8: Sinewave is not clipped by first stage.

Note 9: Current limiting resistor always required.

Supplies

In order to minimize the power dissipation, the negative supply is regulated at -1.9 volts with an internal shunt regulator. This not only reduces warm-up drift of the pole frequencies at power turn-on, but also allows virtually any negative supply greater than -4 volts to be used. The current limiting resistor, R_{EE} , must always be included and is calculated as follows:

$$R_{EE} = \frac{V_{EE} - 2.7V}{0.008}$$

As can be seen from the Block Diagram, an internal 100Ω resistor is in series between the regulator and pin 13. This resistor, which allows for trimming of the control voltage feed-through (explained in further detail below) results in an actual voltage at pin 13 of around -2.7 volts.

Although the circuit was designed for a positive supply of +15 volts, any voltage between +9 and +18 volts may be applied to pin 14. The only effect, other than power dissipation, is the maximum possible peak-to-peak output swing in accordance with:

$$V_{OUT} (\text{V.P.P.}) = V_{CC} - 3V$$

Operation of Each Filter Stage

Each filter stage consists of a variable gain cell followed by a high input impedance buffer. The variable gain cell is a current-in, current-out device (as opposed to the traditional voltage-in, current-out device) whose output current, I_{OUT} , is given by the following expression:

$$I_{OUT} = (I_{REF} - I_{IN}) A_{IO} e^{-V_C/V_T}$$

where $V_T = KT/q$, V_C is the

Absolute Maximum Ratings

voltage applied to pin 12, A_{IO} is the current gain of the cell at $V_C = 0$ (Nominally 0.9), and

$$I_{REF} = \frac{.46V_{CC} - .65V}{100K^*}$$

$\pm 25\%$

As the input to the variable cell is a forward biased diode to ground, it presents essentially a low impedance summing node at a nominal 650mV above ground. The required input currents may therefore be obtained with resistors terminating at this input node.

For normal operation of any filter type, each stage is set up with a feedback resistor, R_F , from the buffer output to the variable gain cell input, and with the pole capacitor, C_P , connected to the output of the variable gain cell. This setup is shown in Figure 1. In the D.C. quiescent state, the buffer output will always adjust itself so that a current equal to I_{REF} flows into the input.

For lowest control voltage feedthrough and maximum peak-to-peak output signal, the quiescent output voltage of each buffer, V_{ODC} , should be:

$$V_{ODC} = .46V_{CC}$$

Thus, in the simple case of Figure 1, R_F is calculated as follows:

$$R_F = \frac{V_{ODC} - .65V}{I_{REF}}$$

$= 100K$ nominal

Since I_{REF} can vary $\pm 25\%$, V_{ODC} can vary nearly 30% from device to device using a standard 5% resistor for R_F . In the typical case where $V_{CC} = +15V$, I_{REF} is $63\mu A$ nominal, and the D.C. output of each buffer should be set for $+6.9V$ nominal.

Voltage Between V_{CC} and V_{EE} Pins	+22V, -0.5V
Voltage Between V_{CC} and Ground Pins	+18V, -0.5V
Voltage Between V_{EE} and Ground Pins	-4V, +0.5V
Voltage Between Cell Input and Ground Pins	+0.5V, -6V
Voltage Between Frequency Control and Ground Pins	$\pm 6V$,
Voltage Between Resonance Control and Ground Pins	+2V, -18V
Current Through Any Pin	$\pm 40mA$
Storage Temperature Range	-55°C to +150°C
Operating Temperature Range	-25°C to +75°C

The output impedance of the variable gain cell, although high, has a finite value. This impedance is reflected back to the input as an A.C. resistance of nominally 1 megohm in parallel with the feedback resistor, R_F , regardless of control voltage value. The pole frequency of each filter section is determined by the total equivalent feedback resistance, R_{EQ} , and the pole capacitor in the expression:

$$f_P = \frac{A_{IO}}{2\pi R_{EQ} C_P} e^{-V_C/V_T}$$

where:

$$R_{EQ} = \frac{R_F \cdot 1M\Omega^*}{R_F + 1M\Omega^*}$$

$\pm 50\%, +100\%$

If the signal is from the output of a previous filter section, it will have a quiescent level of $.46V_{CC}$ (6.9 volts for a +15 volt supply). Therefore, part of I_{IN} will be supplied by this voltage through R_C while the remainder will be sourced through R_F .

The voltage gain in the passband is given by R_{EQ}/R_C . In general, this gain should be set to unity for stages two, three and four. The input resistor to stage one can be scaled for any size of the external input signal. The resistance value should be selected so that the maximum external input signal produces the maximum passband output signal before clipping.

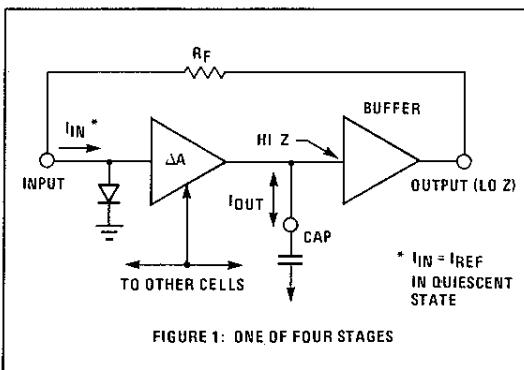


FIGURE 1: ONE OF FOUR STAGES

To generate the hi-pass function, the input signal is coupled into the variable gain element output via the pole capacitor, C_P . Therefore, any D.C. voltage level is blocked by the capacitor and I_{IN} equal to I_{REF} for each input is supplied only through the feedback resistors. The voltage gain in the pass band is simply unity, regardless of the value of R_F . For best results, the output impedance of whatever is generating the external input signal to stage one should be low compared to $R_F/4$.

Sample Filter Circuits

The Block Diagram shows the external components connections for a four-pole, low-pass filter designed to operate off ± 15 volt supplies. The values for R_F , R_C , and R_B were chosen so that a) when the 1 megohm reflected resistance is in parallel with R_F , the gain of stages two, three and four is unity, and b) with the buffer outputs at the proper quiescent level of 6.9 volts, the total current into each input is the required $63\mu A$. For stage 1, all of this quiescent current is sourced by the feedback resistor. For stages two, three, and four, $63\mu A$ is sourced by the feedback resistor, while $70\mu A$ is sourced by the coupling resistor for a total sourced current of $133\mu A$. Thus, to end up with a net quiescent input current of $63\mu A$, $70\mu A$ is sunk out of the input by bias resistor, R_B .

If connecting the filter input to an external signal causes the D.C. level of the filter output to change more than several volts, it is recommended that an input coupling capacitor be used such as shown in Figure 4.

Figures 2, 3, 4, and 5 show high-pass, band-pass, all-pass, and state variable realizations, all with the voltage controlled

resonance feature. Note that due to the configuration of the resonance feedback, the resonance frequency of the high-pass will be approximately 2.4 times higher than that of the low-pass, while the resonance frequency of the band-pass and all-pass will be $1/2.4 = .42$ times lower than that of the low-pass, for the same component values. For the state variable, resistor R_Q adds positive feedback to increase the maximum Q, which is otherwise limited by the reflected $1M\Omega$ impedance across the integrators.

Pole Frequency Control Scale

The current gains of each of the four sections (and consequently their pole frequencies) are controlled simultaneously with a voltage applied to pin 12. Since the scale is exponential with the standard $18mV/octave$ ($60mV/decade$), an input attenuator network will in most cases be required. An increasing positive control voltage lowers the pole frequencies of the filter. For best results over a thousand-to-one control range, the voltage on pin 12 should be maintained between $-25mV$ and $+155mV$.

Unlike the typical variable transconductance cell used in most V.C. filters, the four stages in the CEM 3320 are fully temperature compensated. The only remaining first order temperature effect is that of control scale sensitivity ($1/V_T$). This effect may be compensated in the usual manner with a $+3300ppm$ tempco resistor (Tel Labs Q81).

Resonance Control

The variable gain cell used to control the amount of resonance is the traditional transconductance type of amplifier. It has a separate signal voltage input

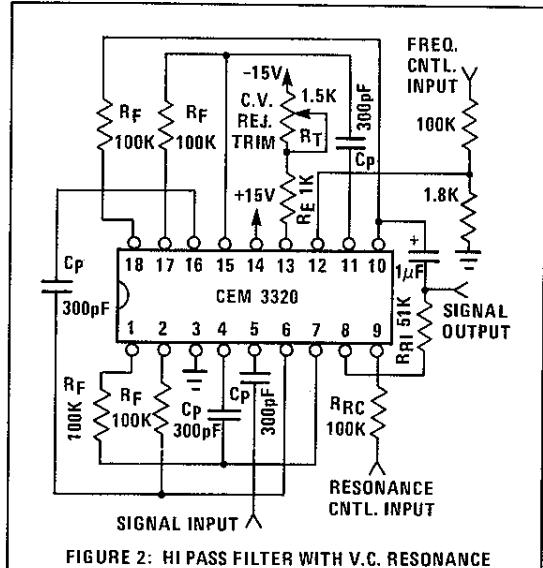


FIGURE 2: HI PASS FILTER WITH V.C. RESONANCE

(pin 8), a separate control current input with a modified linear scale (pin 9), and a current output internally connected to the input of stage one. With an impedance of $3.6K \pm 900\Omega$, the input is referenced to ground; thus, connection to the filter output will require a coupling capacitor.

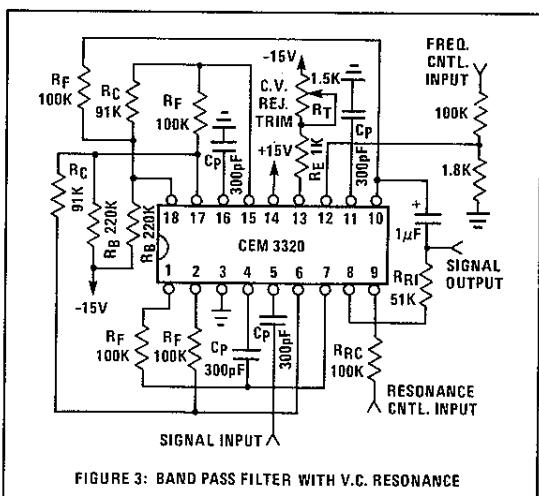


FIGURE 3: BAND PASS FILTER WITH V.C. RESONANCE

Control of the transconductance is accomplished with a current input. As the control input is a low impedance summing node at a potential near ground, the control current may be derived from the resonance control voltage with an input resistor, R_{RC} , terminated at pin 9. This resistor should be selected so that the maximum available resonance control voltage produces the maximum desired control current.

Figure 6 shows a graph of the transconductance versus control current. As can be seen, the slope of the curve becomes more gradual as the control current increases. This feature allows the resonance to be controlled with finer resolution as the critical point of oscillation is approached.

The maximum control current is therefore selected in accordance with the amount of control sensitivity which is desired at the top of the control range. The value of the input resistor, R_{RI} , is then selected depending on where in the control scale oscillation is desired to begin (when the control voltage is 90% of the maximum value, for instance). The following formula may be used:

$$R_{RI} = 3.6K * \left(\frac{G_{MOSC} R_{EQ} - 1}{A_{OSC}} \right) * \pm 25\%$$

where G_{MOSC} is the transconductance corresponding to the control current at which oscillation is desired to begin; and where A_{OSC} is the overall gain from the resonance signal input resistor, R_{RI} , to the filter output required to sustain oscillation. If the gain of stages 2, 3 and 4 are unity, then $A_{OSC} = 12dB$ or 4 in the case of the low pass filter.

While operating the filter in the resonant mode, care should be taken not to overload the input to the filter. If the signal output of stage one is allowed to become clipped, then not only will the apparent resonance of the signal at the filter output appear to be reduced, but the D.C. level of the output signal will shift.

When the resonance control is advanced until sustained oscillations are produced, advancing the resonance control further will merely increase the amplitude of the oscillation. A lesser effect is the shift of the oscillation frequency. For minimum shift (typically less than 0.5%), the oscillation amplitude should be kept below the clipping level of the first stage output. Allowing the oscillation to be clipped will produce frequency shifts in excess of 5%.

Other Uses of the Resonance Control Cell

Other than controlling the resonance, the variable transconductance amplifier may be used as an independent VCA controlling the amplitude of the input signal to the filter. Or the cell may be set up as a symmetrical limiter/clipper for either preventing large dynamic input signals from overloading the filter or for providing additional coloration to the input signal.

Pole Frequency Control Voltage Rejection

The D.C. voltage shift at the filter output due to the frequency control voltage may be minimized by adjusting the current into the minus supply pin, pin 13. This is accomplished by replacing the negative supply current limiting resistor, R_{EE} ,

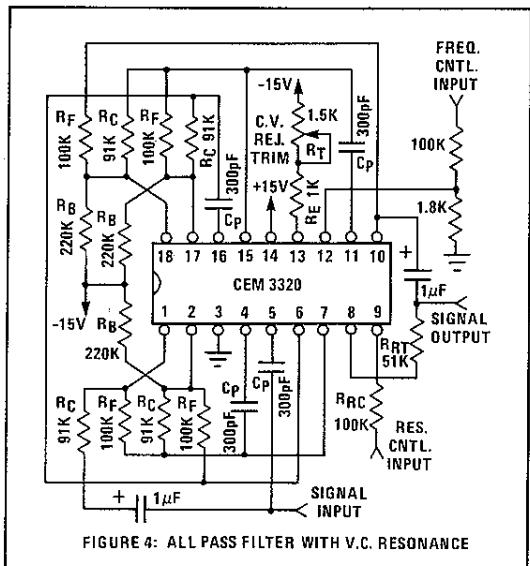


FIGURE 4: ALL PASS FILTER WITH V.C. RESONANCE

with a series resistor and trim pot. The fixed resistor, R_E , and series trim pot, R_T , should be selected so that the current into pin 13 may be adjusted from 5mA to 12mA. Or:

$$R_E = \frac{V_{EE} - 3.2V}{12mA}$$

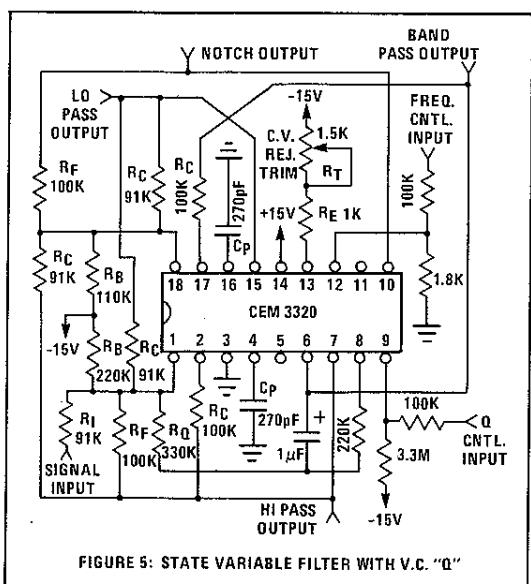


FIGURE 5: STATE VARIABLE FILTER WITH V.C. "Q"

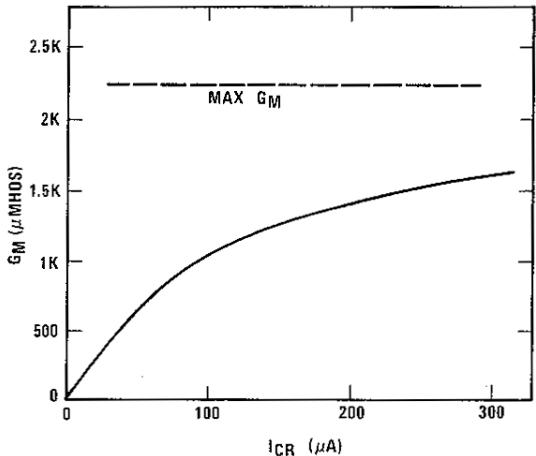


FIGURE 6: TRANSCONDUCTANCE V.S. CONTROL CURRENT OF RESONANCE CELL

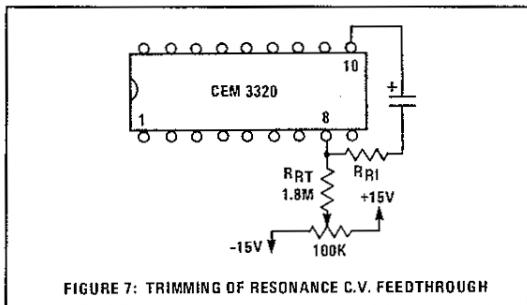


FIGURE 7: TRIMMING OF RESONANCE C.V. FEEDTHROUGH

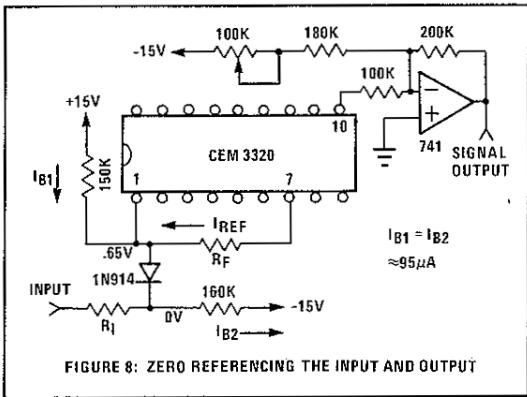


FIGURE 8: ZERO REFERENCING THE INPUT AND OUTPUT

and

$$R_T = \frac{V_{EE} - 2.4V}{5mA} - R_E$$

These components are shown in the filter circuits of Figures 2-5. To obtain minimum control voltage feedthrough, the best technique for adjusting this trim is to switch back and forth between the maximum and minimum control voltages while adjusting the pot so that the D.C. output voltage at these two extreme conditions is the same.

Resonance Control Voltage Rejection

For most applications, no trimming should be necessary. However, if required, the resonance control voltage feedthrough may be minimized by applying a small D.C. voltage on the resonance signal input pin, pin 8. A typical set-up is shown in Figure 7. The value of R_{RT} should be selected so the trim pot is able to adjust the voltage on pin 8 by $\pm 30mV$.

Stage Buffers

Each buffer can source up to 10mA and sink a nominal 500 μ A. However, any D.C. load greater than $\pm 200\mu$ A to $\pm 300\mu$ A

may begin to degrade the performance of the filter, especially if the loads on each buffer differ by more than this amount. The maximum recommended D.C. loads are 1mA source, 250 μ A sink, and a 150 μ A load difference between buffers. The maximum recommended A.C. loads are $\pm 250\mu$ A.

Since the D.C. level at the filter output is at some non-zero voltage (6.9 volts for $V_{CC} = +15V$), a coupling capacitor will be required somewhere in the signal chain, either at the filter output or the following device inputs. Note that if the resonance feature is being used, the filter output is already D.C. blocked by the resonance input coupling capacitor, thus providing a convenient output point. If D.C. coupling to ground referenced inputs and outputs is required, the schemes shown in Figure 8 may be used. Note that the output circuit has the benefits of 1) allowing for gain after the filter, and 2) providing an output with greater drive capability. The buffer outputs are not short circuit protected; therefore care should be exercised to not short the outputs to ground or either supply.

