

Rapport de test

Synthlab

Chevallier Raphaël
Chevalot Emilie
Guerra Elia
Hardouin Clément
Kwete Dany
Le Hô Romain

Sommaire

[Tests unitaires](#)

[Technologies employées](#)

[Test des abstractions](#)

[Test des filtres](#)

[Test des modules](#)

[Autres tests](#)

[Wire](#)

[Synthesizer](#)

[Test des contrôleurs](#)

[Test des modules](#)

[Test des ports](#)

[Autres tests](#)

[Tests du synthétiseur](#)

[Tests de wire](#)

[Test des commandes](#)

[Test de la factory](#)

[Test des utils](#)

[Tests fonctionnels](#)

Préface

But

Ce document permet de vous présenter les différentes façons dont nous avons tester notre code.

Audience

Ce document s'adresse au personnel et étudiants de l'ISTIC.

Tests unitaires

Technologies employées

Nous avons utilisé JUnit pour effectuer ces tests unitaires mais aussi Eclemma afin de visualiser notre couverture de code et ainsi pouvoir apporter des améliorations si besoin.

Tests généraux

Nom du test	Description du test	OK / KO
testSetGetFoo	Vérifie que si on donne une valeur à foo, getFoo() retourne cette valeur	OK

Test des abstractions

Test des filtres

Nom du test	Description du test	OK / KO
testGenerateIntInt (classe AmplitudeModulatorFilterTest)	Vérifie si le port de sortie correspond au produit du port d'entrée par le port qui module l'amplitude	OK
testGenerateIntInt (classe AttenuationFilterTest)	Vérifie si le port de sortie correspond au produit du port d'entrée par la valeur de l'atténuation	OK
testGenerateIntInt (classe FrequencyModulatorFilterTest)	Vérifie si le port de sortie correspond au port d'entrée modulé par la fréquence	OK

testGenerateIntInt (classe QuadrupleMixFilterTest)	Vérifie si le port de sortie est égal à la somme des ports d'entrée)	OK
testGenerateIntInt (classe TriplePassThroughFilterTest)	Vérifie si les ports de sorties sont tous égaux au port d'entrée	OK

Test des modules

Nom du test	Description du test	OK / KO
testGetJSyn (tous les modules)	Vérifie si getJSyn() est non null	OK
testGetWires (tous les modules sauf OUT)	Vérifie si getWires() retourne bien tous les câbles reliés à un module (tous les ports des câbles appartiennent au module)	OK
testGetWiresDifferent (tous les modules)	Vérifie si getWires() retourne bien tous les câbles reliés à un module (un des câbles n'appartient pas au module)	OK
testGetWiresDifferentBad (tous les modules)	Vérifie si getWires() lance une exception lorsqu'on connecte deux câbles différents au même port d'entrée	OK
testGetWiresNotConnected (tous les modules)	Vérifie qu'au début il n'y a aucun câble	OK
testGetModel (module AudioScope)	Vérifie si le modèle retourné est celui du scope de JSyn	OK
testSetGetDecayNegative (module EG)	Vérifie que si l'on donne une valeur négative à decay, getDecay() retourne 0.0	OK

testSetGetAttackNegative (module EG)	Vérifie que si l'on donne une valeur négative à attack, getAttack() retourne 0.0	OK
testSetGetSustainNegative (module EG)	Vérifie que si l'on donne une valeur négative à sustain, getSustain() retourne 0.0	OK
testSetGetReleaseNegative (module EG)	Vérifie que si l'on donne une valeur négative à release, getRelease() retourne 0.0	OK
testGetWires {numero du port d'entrée } (module MIX)	Équivalent de testGetWires pour les 4 ports d'entrée du module MIX	OK
testSetGetAttenuation {numero du port d'entrée} (module MIX)	Vérifie que si on donne une valeur à l'atténuation d'un des 4 ports d'entrée, getAttenuation{ numéro du port d'entrée}() retourne cette valeur	OK
testSetIsMuteFalse(module Out)	Vérifie que si on donne la valeur false à mute, isMute() retourne false	OK
testSetIsMuteTrue(module Out)	Vérifie que si on donne la valeur true à mute, isMute() retourne true	OK
testUpdate (modules VCA,VCF_HP,VCF_LP,VCO)	Vérifie si on tient compte du fait qu'un câble est connecté à un port de modulation lorsqu'on met à jour le module	OK
testUpdateWithoutConnect (modules VCA,VCF_HP,VCF_LP,VCO)	Vérifie si on tient compte du fait qu'aucun câble est connecté à un port de modulation lorsqu'on met à jour le module	OK

testSetGetOctave (module VCO)	Vérifie que si l'on donne une valeur à l'octave du module, getOctave() retourne cette valeur et frequency est modifié en conséquence	OK
testSetGetTone (module VCO)	Vérifie que si l'on donne une valeur à tone du module, getTone() retourne cette valeur et frequency est modifié en conséquence	OK

Autres tests

Wire

Nom du test	Description du test	OK / KO
testGetInput	Vérifie si le port d'entrée du câble est null au début	OK
testGetOutput	Vérifie si le port de sortie du câble est null au début	OK
testIsConnected	Vérifie si le câble n'est pas connecté au début	OK
testConnectInputPort	Vérifie si la connexion du port d'entrée se fait bien	OK
testConnectOutputPort	Vérifie si la connexion du port de sortie se fait bien	OK
testDisconnect	Vérifie si la déconnexion du câble se fait bien	OK

Synthesizer

Nom du test	Description du test	OK / KO
testGetJSyn	Vérifie si getJSyn() est non null	OK

testAddModule	Vérifie si AddModule() ajoute un module au synthétiseur	OK
testRemoveModule	Vérifie si RemoveModule() enlève un module au synthétiseur	OK
testAddWire	Vérifie si AddWire() ajoute un câble au synthétiseur	OK
testRemoveWire	Vérifie si RemoveWire() enlève un câble au synthétiseur	OK
testIsRunning	Vérifie si le synthétiseur n'est pas lancé au début	OK
testStart	Vérifie si start() démarre le synthétiseur	OK
testStop	Vérifie si stop() arrête le synthétiseur	OK
testStartModule	Vérifie si le fait de démarrer un module ne démarre pas le synthétiseur	OK
testStopModule	Vérifie si le fait d'arrêter un module n'arrête pas le synthétiseur	OK
testSetGetCurrentWire	Vérifie que si on donne une valeur à currentWire, getCurrentWire() retourne cette valeur	OK

Test des contrôleurs

Test des modules

Les modules étant tous basés sur le même principe, les fonctions des modules vont juste différer lorsqu'il y a des paramètres.

Nom du test	Description du test	OK / KO
testGetPresentation	getPresentation doit renvoyer un objet instancié	OK
testP2cClosingWireNotConnected	Il ne doit pas y avoir d'erreur lorsqu'on demande l'on enlève les câbles alors qu'il n'y en a pas	OK
testP2cClosingAllWireConnected	Il ne doit pas avoir de soucis lorsque l'on déconnecte les câbles connecté. Le test connecte tout les inputs et outputs sur des câbles différents	OK
testGetSynthesizerPresentation	getSynthesizerPresentation doit renvoyer un objet instancié	OK
testP2cRemoveModule	Le module enlevé doit être retiré des modules présents dans le synthétiseur	OK
testP2cFooChangedOnce	Vérifie que le paramètre foo soit bien modifié lorsqu'on le change	OK
testP2cFooChangedTwice	Vérifie que le paramètre foo soit bien modifié lorsqu'on le change deux fois	OK
testP2cFooChangedLessMin	Vérifie que le paramètre foo ne puisse pas être affecté à une valeur inférieure à son minima	OK
testP2cFooChangedMoreMax	Vérifie que le paramètre foo ne puisse pas être affecté à une valeur supérieur à son maxima	OK

testSetParameter	Vérifie que lorsqu'on demande à setParameter de modifier un paramètre ce dernier est effectivement modifié	OK
testSetParameterFalseParameter	Vérifie que aucun paramètre ne soient modifié lorsque le nom donné en argument est faux	OK

Les tests vérifiant les minima nous ont montré qu'il manquait cette vérification. Nous l'avons par conséquent corrigé.

Test des ports

Nom du test	Description du test	OK / KO
testCInputPortString	Vérifie si la présentation est non null et si le constructeur modifie bien le nom du CInputPort	OK
testCInputPortUnitInputPort	Vérifie si la présentation est non null et si getJSyn retourne l'input port (de JSyn) et si le constructeur modifie bien le nom du CInputPort	OK
testCOutputPortString	Vérifie si la présentation est non null et si le constructeur modifie bien le nom du COutputPort	OK
testCOutputPortUnitOutputPort	Vérifie si la présentation est non null et si getJSyn retourne l'output port (de JSyn) et si le constructeur modifie bien le nom du COutputPort	OK

Autres tests*Tests du synthétiseur*

Nom du test	Description du test	OK / KO
testAddModule	Vérifie que la fonction addModule ajoute bien un module dans la liste	OK
testGetPresentation	getPresentation doit renvoyer un objet instancié	OK
testP2cAddModule	Même chose que pour testAddModule	OK
testP2cRemoveModule	Le module enlevé doit être retiré des modules présents dans le synthétiseur	OK

Tests de wire

Nom du test	Description du test	OK / KO
testDisconnect	Vérifie que le câble soit bien déconnecté	OK
testGetPresentation	getPresentation doit renvoyer un objet instancié	OK
testGetCurrentWireColor	Vérifie que la fonction getCurrentWireColor ne renvoie pas d'objet null	OK
testGetSynthesizerPresentation	getSynthesizerPresentation doit renvoyer un objet instancié	OK

Test des commandes

Nom du test	Description du test	OK / KO
testExecute (classe AddModule{nom du module}CommandTest)	Vérifie si le module {nom du module} est bien ajouté aux modules du synthétiseur	OK
testExecute(classe NewSynthCommandTest)	Vérifie si un nouveau synthétiseur est créé	OK
testExecuteSame (classe OpenAndSaveCommandTest)	On sauvegarde un montage, on le réinitialise, on l'ouvre. Le montage obtenu doit être celui sauvegardé (les fichiers ouverts et sauvegardés doivent être les mêmes. Si ce n'est pas le cas, on demandera à l'utilisateur de choisir les fichiers	OK
testExecuteModifySame (classe OpenAndSaveCommandTest)	Comme testExecuteSame mais après avoir sauvegardé le montage, on le modifie, on le sauvegarde à nouveau puis on le réinitialise et on l'ouvre. Le montage obtenu doit être celui sauvegardé	OK
testExecute (classe ToolbarCurrentWireColorTest)	Vérifie si le couleur du câble courant est bien modifiée) c-à-d si la couleur de Synthesizer est celle de SynthFrame	OK
testExecutePause(classe PausePlayCommandTest)	Vérifie si le synthétiseur s'est bien arrêté	OK
testExecutePlay(classe PausePlayCommandTest)	Vérifie si le synthétiseur a bien démarré	OK

Test de la factory

Nom du test	Description du test	OK / KO
testNew{Objet à créer} (classes AFactoryTest, PFactoryTest et CFactoryTest)	Vérifie si l'objet retourné est du type {Objet à créer}	OK
testSetGetAFactory(classe PACFactoryTest)	Vérifie si une AFactory est bien modifiée et retournée	OK
testSetGetCFactory(classe PACFactoryTest)	Vérifie si une CFactory est bien modifiée et retournée	OK
testSetGetPFactory(classe PACFactoryTest)	Vérifie si une PFactory est bien modifiée et retournée	OK
testSetGetFactoryNull(clas se PACFactoryTest)	Vérifie si une AFactory est retournée lorsque cFactory est null	OK
testSetGetFactoryNotNull(classe PACFactoryTest)	Vérifie si une CFactory est retournée lorsque cFactory n'est pas null	OK

Test des utils

Nom du test	Description du test	OK / KO
testSaveAndLoadSetTone	On ajoute des modules à un synthétiseur dont un moduleVCO pour lequel on modifie la valeur de tone. On sauvegarde ce synthétiseur. On le réinitialise. Lorsqu'on ouvre ensuite le fichier de sauvegarde on doit obtenir le montage sauvegardé précédemment	OK
testSaveAndLoadSetAtten uation	Idem que testSaveAndLoadSetTone sauf qu'on modifie la valeur de l'atténuation du VCA	OK

testSaveAndLoadWire	Idem que testSaveAndLoadSetTone sauf qu'on ajoute des câbles	OK
testSaveAndLoadWireVCF	Idem que testSaveAndLoadWire mais le câble est relié à un VCF	OK

Tests fonctionnels

Pour les tests fonctionnels, nous avons crée différents montage pour essayer de mettre à mal l'application. Ceci nous a permis de corriger quelques dysfonctionnements.